Version 11 Release 0
May 2018

IBM Marketing Platform Administrator's Guide

Note
Before using this information and the product it supports, read the information in "Notices" on page 623.

This edition applies to version 11, release 0, modification 0 of IBM Marketing Platform and to all subsequent releases and modifications until otherwise indicated in new editions.

© Copyright IBM Corporation 1999, 2018.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Chapter 1. Introduction to IBM Marketing Platform. 1

About Marketing Platform security features 1
Configuration management 3
Localization in IBM Marketing Software 3
The common user interface 4
Logging in to IBM Marketing Software 4
Marketing Platform documentation and help . . . 5

Chapter 2. IBM Marketing Software user account management 7

Types of user accounts: internal and external . . . 7
Properties of internal user accounts 8
Adding internal user accounts 9
Deleting internal user accounts 9
Changing internal user password expiration dates . . 9
Resetting internal user passwords 10
Changing internal user account properties 10
Changing internal user system status 10
Adding internal user data sources 11
Changing internal user data sources 11
Deleting internal user data sources 11
The user management pages. 12
Locale preference 13
Setting the user locale preference 14
Synchronization of external users 14
Forcing synchronization of external users . . . 14

Chapter 3. Security management . . . 15

Permissions and tasks of the security administrator in Marketing Platform 15
Special characters in role and policy names. . . . 16
Roles and permissions in Marketing Platform and Campaign 16
Overview of managing user application access in Marketing Platform. 16
Types of groups: internal and external 17
Partitions and security management 18
Pre-configured users and roles 19
Cross-partition administration privileges. 20
Adding an internal group 21
Adding a subgroup. 21
Deleting a group or subgroup 21
Changing a group or subgroup description 22
Assigning a group to a partition 22
Adding a user to a group or subgroup 22
Removing a user from a group or subgroup . . . 23
The user group management pages 23
Creating a role 25
Modifying role permissions 25
Removing a role from the system 26
Assigning a role to or removing a role from a group 26
Assigning a role to or removing a role from a user 27
Definitions of permission states. 27
Permissions for products that use only basic roles 28
Permissions for Marketing Platform 29
Permissions for Opportunity Detect 30

Chapter 4. Configuration management 33

Property categories 33
Category types 33
Templates for duplicating categories 34
Category naming restrictions 34
Property descriptions 34
The refresh function 35
The default user locale preference 35
Navigating to a category 35
Editing property values 36
Creating a category from a template 36
Deleting a category 36

Chapter 5. Dashboard management . . 39

Dashboard planning 39
Dashboard audiences 39
User permissions required to view dashboards . . 40
Pre-defined portlets. 40
Pre-defined portlet availability 40
Marketing Operations IBM Cognos report portlets 40
Marketing Operations list portlets 41
IBM Cognos report portlets for Campaign . . . 42
Campaign list portlets 43
eMessage IBM Cognos report portlets 43
Interact IBM Cognos report portlet 43
Distributed Marketing list portlets 44
Contact Optimization list portlets 44
Pre-assembled dashboards 45
Pre-assembled dashboard availability 45
List of pre-assembled dashboards 45
IBM Cognos report performance considerations . . 46
Scheduling a dashboard report 47
Dashboard setup 47
Permissions required to administer dashboards 48
Dashboard layout 48
Dashboards and partitions 48
Overview of working with dashboards in a multi-partition environment 49
Enabling or disabling pre-defined portlets . . . 49
Creating a dashboard that is not pre-assembled 50
Creating a pre-assembled dashboard 50
Adding a pre-defined portlet to a dashboard . . 51
Removing a portlet from a dashboard 51
Changing the name or properties of a portlet . . 51
Changing the name or properties of a dashboard 52
Deleting a dashboard 52
Assigning or changing a dashboard administrator 52
The Manage Portlets page 53
Quick link portlets 53
Creating a quick link portlet. 54
Custom portlets 54

© Copyright IBM Corp. 1999, 2018

Custom portlet types and availability 54
Authentication considerations for custom portlets 54
Overview of the portlet creation process 55
Preparing the URL from a Digital Analytics for On Premises report 55
Preparing the URL from an IBM Cognos dashboard report 56
Preparing the URL from a Digital Analytics report 57
Preparing the URL from an intranet or internet page 57
Adding a custom portlet to a dashboard 57
Dynamic tokens 58
The Create Custom Portlet page 59
Dashboard membership administration 59
The dashboard administrator 60
Granting or removing dashboard membership . . 60

Chapter 6. The IBM Marketing Software Scheduler 61
Scheduler triggers that are sent on success or failure of runs 62
Schedules that depend on completion of multiple runs 62
Schedule triggers that are sent from an external script 63
Scheduler recurrence patterns 64
Time zone support 65
Scheduler throttling 65
Setting up throttling for the IBM Marketing Software Scheduler 66
Whitelist prerequisite for external tasks (with FixPack 10.0.0.1 only) 67
Adding a script to the whitelist 67
Adding an API to the whitelist 68
Best practices for setting up schedules 68
The create a schedule wizard 69
Run exclusions 73
Viewing exclusion rules 73
Adding exclusion rules 73
Deleting exclusion rules 74
Enabling and disabling exclusion rules 75
Importing exclusion rules 75
What to consider when you use the scheduler with Campaign 76
Difference between the IBM Campaign Schedule process and IBM Marketing Software Scheduler . 77
Permissions for scheduling flowcharts 77
Creating a flowchart schedule using default parameters 78
About overriding the default parameters for Campaign flowchart run schedules 78
Schedule notifications 79
Setting up notifications for schedules you create 80
Deleting or modifying notifications for schedules you create 80
Setting up schedule notifications for a group of users 80
The My job notifications page 81
Schedule management 81
The Schedule management pages 82

Chapter 7. SAML 2.0 based federated authentication 87
How to implement federated authentication . . . 89
Creating the data repository 89
Configuring the IdP data source in the web application server 91
Setting up the classpaths for the IBM IdP client façade 91
Deploying the IdP server 92
Configuring the IdP server 92
Obtaining keystores and importing them into the IdP server 94
Setting configuration properties on the Configuration page 94
Onboarding Service Providers and users 95
Using the IdP client façade to generate tokens and pass them to Service Providers 95
Related concepts 96

Chapter 8. SAML 2.0 single sign-on . . 99
Behavior when SAML 2.0 single sign-on is implemented 100
Configuration process roadmap: SAML 2.0 single sign-on 101
Setting up the metadata file 101
Setting SAML 2.0 configuration properties 102
Setting up a data source for SAML single sign-on 102
Sample SAML 2.0 IdP assertion 102
Sample IdP metadata 104

Chapter 9. Configuring JWT authentication between applications . 107

Chapter 10. Single sign-on between IBM Marketing Software and IBM Digital Analytics 109
Setting up single sign-on between IBM Marketing Software and Digital Analytics using automatic user account creation 110
Setting up single sign-on between IBM Marketing Software and Digital Analytics using manual user account creation 111
Configuring WebLogic for single sign-on between Digital Analytics and IBM Marketing Software . . 113
Configuring WebSphere for single sign-on between Digital Analytics and IBM Marketing Software . . 113
Digital Analytics integration with Websense using a custom proxy 114
Deploying the custom proxy on WebSphere . . 114
Importing the Digital Analytics certificate when WebSphere does not have outbound access . . 115
Importing the Digital Analytics certificate when WebSphere has outbound access 115
Next steps 116

Chapter 11. Integration between IBM Marketing Software and Windows Active Directory 117
Active Directory integration features 117

Active Directory integration prerequisites 120
Configuration process roadmap: Active Directory integration 120
Obtaining required information 121
Group membership, mapping, and application access 122
Storing directory server credentials in Marketing Platform 123
Setting LDAP login method connection properties in IBM Marketing Software 124
Setting LDAP synchronization properties . . . 124
Setting user attributes map properties 125
Mapping LDAP groups to IBM Marketing Software groups 126
Testing synchronization 126
Setting up an Active Directory user with PlatformAdminRole permissions 126
Setting the security mode to enable NTLMv2 authentication 127
Restarting the web application server 127
Testing login as an Active Directory user . . . 127

Chapter 12. Integration between IBM Marketing Software and LDAP servers 129
LDAP integration features 129
LDAP integration prerequisites 132
Configuration process roadmap: LDAP integration 132
Obtaining required information 132
Group membership, mapping, and application access 134
Storing directory server credentials in Marketing Platform 134
Setting LDAP login method connection properties in IBM Marketing Software 135
Setting LDAP synchronization properties . . . 136
Setting user attributes map properties 136
Mapping LDAP groups to IBM Marketing Software groups 137
Testing synchronization 138
Setting the security mode to LDAP 138
Restarting the web application server 138
Testing login as an LDAP user 138

Chapter 13. Integration with web access control platforms 139
About context roots 140
SiteMinder integration prerequisites 141
Configuring SiteMinder for IBM Marketing Software products 141
Enabling single logouts with SiteMinder . . . 143
IBM Security Access Manager integration prerequisites. 144
Configuring IBM Security Access Manager for IBM Marketing Software products 144
Configuration process roadmap: integrating IBM Marketing Software with a web access control system 147
Performing LDAP integration 147
Setting web access control connection properties in IBM Marketing Software. 147
Restarting the web application server 148
Testing web access control synchronization and IBM Marketing Software login. 148
Configuring integration with an SSL type of WebSEAL junction. 149

Chapter 14. Alert and notification management. 151
Alert and notification subscriptions 151
Setting system alert and notification subscriptions 151
Configuring email notifications in IBM Marketing Software 152

Chapter 15. Implementation of one-way SSL 153
Overview of SSL certificates 153
Client and server roles in IBM Marketing Software 154
SSL in IBM Marketing Software 155
Configuration process roadmap: implementing SSL in IBM Marketing Software. 156
Certificates for SSL 157
Creating and configuring certificates. 157
Creating and configuring certificates for a clustered environment 159
Configuring your web application servers for SSL 162
Ensuring cookie security 162
Configuring IBM Marketing Software for SSL . . 163
Configuring SSL in Marketing Platform . . . 163
Configuring SSL in Platform for a clustered environment. 164
Configuring SSL in Marketing Platform with LDAP integration 164
Configuring SSL in Marketing Platform with data filters 165
Configuring SSL in Marketing Operations . . . 165
Configuring SSL in Campaign 166
Configuring SSL in Campaign for a clustered environment. 167
Configuring IBM Campaign in SSL and IBM Campaign Listener in non-SSL 167
Configuring SSL in Contact Optimization . . . 170
Configuring SSL in Interact. 170
Configuring SSL in Distributed Marketing. . . 170
Configuring SSL in Reports. 171
Configuring SSL in Digital Analytics for On Premises 171
Verifying your SSL configuration 171
Useful links for SSL 171
Quality of protection (QoP) settings for WebLogic 172
Quality of protection (QoP) settings for WebSphere 172

Chapter 16. Security framework for IBM Marketing Software APIs 173

Chapter 17. Data filter creation and management. 177
Overview of data filter creation 177
Data filter associations to restrict user access 177

Two ways to create data filters: automatic generation and manual specification 177
Two ways to assign users and groups: in the user interface and in the XML 178
Data filter concepts 178
Configuration process roadmap: creating data filters 179
Planning your data filter criteria: automatic generation 179
Planning your data filter criteria: manual generation 180
Obtaining the JDBC driver for your database: automatic generation only 180
Obtaining required information 181
Creating the XML to specify data filters . . . 181
Setting required data filter configuration properties 182
Populating the data filter system tables 183
Assigning users and groups to data filters . . . 183
Data filter XML reference 183
About the IDs in the XML 183
AddDataConfiguration | dataConfiguration . . 184
AddLogicalFields | logicalFields | LogicalField 184
GenerateDataFilters 184
GenerateDataFilters | fixedFields | FixedField 185
GenerateDataFilters | profileField | ProfileField 186
AddDataTable | dataTable 186
AddDataFilters | dataFilters | DataFilter . . . 186
AddDataFilters | dataFilters | DataFilter | fieldConstraints | FieldConstraint 187
AddDataTable | dataTable | fields | TableField 187
AddAudience | audience 187
AddAudience | audience | fields | AudienceField 187
addAudienceTableAssociations | addAudienceTableAssociation | audienceTableAssociation 188
AddAssignments | assignments | AssignmentByName 188
Example: Manually specifying data filters 189
Example: Automatically generating a set of data filters 193
About assigning users and groups in the XML . . 198
About assigning user and groups though the user interface 203
Advanced search 203
Viewing assigned data filters 204
Assigning users and groups to data filters . . . 204
Removing data filter assignments 205
Adding data filters after the initial set has been created 205

Chapter 18. Audit event tracking in IBM Marketing Software 207

Limitations on audit event tracking 207
Legacy audit events 207
Retroactive changes 208
Permissions for viewing the Audit Events report in a multi-partition environment 208
Enabling and disabling event auditing 208
Configuring which audit events appear in the report 208
Modifying the audit report content and display 209
Fields in the Report Parameters window 210
Fields and buttons in the Audit Events report . . 211
Archived audit events 211
Configuring audit backup notifications 212
Exporting the Audit Events report 213
Optimizing the export of the Audit Events report for large event volumes 213

Chapter 19. The Marketing Platform system log 215

System log configuration 215
Enabling single-user logging 216

Chapter 20. IBM Marketing Platform utilities and SQL scripts 219

Setting up Marketing Platform utilities on additional machines 221
Marketing Platform utilities 221
alertConfigTool 221
configTool 222
datafilteringScriptTool 226
encryptPasswords 227
partitionTool 228
populateDb 231
restoreAccess 231
scheduler_console_client 233
Marketing Platform SQL scripts 235
ManagerSchema_DeleteAll.sql 235
ManagerSchema_PurgeDataFiltering.sql . . . 236
ManagerSchema_DropAll.sql 236
SQL scripts for creating system tables 237

Chapter 21. Configuration properties on the Configuration page 239

IBM Marketing Platform configuration properties 239
IBM Marketing Platform 239
IBM Marketing Software | General | Navigation 241
IBM Marketing Software | General | Data filtering 242
IBM Marketing Software | General | Password settings 243
IBM Marketing Software | General | Miscellaneous 246
IBM Marketing Software | General | Communication | Email 246
IBM Marketing Platform | Scheduler 248
IBM Marketing Platform | Scheduler | Recurrence definitions 249
IBM Marketing Platform | Scheduler | Schedule registrations | [Product] | [Object type] 251
IBM Marketing Platform | Scheduler | Schedule registrations | [Product] | [Object type] | [Throttling group] 251
IBM Marketing Platform | Security 252

IBM Marketing Platform | Security | Login method details | LDAP 252
IBM Marketing Platform | Security | Login method details | Web access control. 255
IBM Marketing Platform | Security | Login method details | SAML 2.0. 256
IBM Marketing Platform | Security | LDAP synchronization 260
IBM Marketing Platform | Security | LDAP synchronization | LDAP reference to IBM Marketing Platform group map 268
IBM Marketing Platform | Security | Federated authentication 269
IBM Marketing Platform | Security | Federated authentication | partitions | partition[n] . . . 270
IBM Marketing Platform | Security | API management 270
IBM Marketing Platform | Security | API management | [Product] | (API configuration template) 271
IBM Marketing Platform | Security | JWT authentication 272
IBM Marketing Platform | Notifications . . . 273
IBM Marketing Platform | Audit Events . . . 274
IBM Marketing Platform | Audit Events | Audit events configuration 274
IBM Marketing Platform | Audit Events | Audit events severity configuration 277
Digital Analytics configuration properties 277
Digital Analytics 277
Digital Analytics | Integration | partitions | partition[n] 278
Report configuration properties 278
Reports | Integrations | Cognos [version]. . . 278
Reports | Schemas | [product] | [schema name] | SQL Configuration 282
Reports | Schemas | Campaign 283
Reports | Schemas | Campaign | Offer Performance. 283
Reports | Schemas | Campaign | [schema name] | Columns | [Contact Metric] and [Response metric] 284
Reports | Schemas | Campaign | Campaign Performance. 287
Reports | Schemas | Campaign | Campaign Offer Response Breakout 288
Reports | Schemas | Campaign | Campaign Offer Response Breakout | Columns | [Response Type] 288
Reports | Schemas | Campaign | [schema name] | Columns | [Response Metric] 289
Reports | Schemas | Campaign | Campaign Offer Contact Status Breakout 291
Reports | Schemas | Campaign | Campaign Offer Contact Status Breakout | Columns | [Contact Status] 291
Reports | Schemas | Campaign | Campaign Custom Attributes | Columns. 292
Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Campaign Custom Column] 295
Reports | Schemas | Campaign | [schema name] | Columns | [Contact Metric] 295
Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Offer Custom Column] 297
Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Cell Custom Column] 297
Reports | Schemas | Interact 298
Reports | Schemas | Interact | Interact Performance. 299
Reports | Schemas | eMessage 300
IBM Marketing Operations configuration properties 300
Marketing Operations 300
Marketing Operations | navigation 301
Marketing Operations | about. 303
Marketing Operations | umoConfiguration . . 304
Marketing Operations | umoConfiguration | Approvals 308
Marketing Operations | umoConfiguration | templates. 310
Marketing Operations | umoConfiguration | attachmentFolders. 311
Marketing Operations | umoConfiguration | fileUpload 313
Marketing Operations | umoConfiguration | Email 314
Marketing Operations | umoConfiguration | markup 314
Marketing Operations | umoConfiguration | grid 316
Marketing Operations | umoConfiguration | workflow. 317
Marketing Operations | umoConfiguration | integrationServices 318
Marketing Operations | umoConfiguration | campaignIntegration 319
Marketing Operations | umoConfiguration | reports 319
Marketing Operations | umoConfiguration | invoiceRollup 320
Marketing Operations | umoConfiguration | database 321
Marketing Operations | umoConfiguration | listingPages 324
Marketing Operations | umoConfiguration | objectCodeLocking 325
Marketing Operations | umoConfiguration | thumbnailGeneration. 326
Marketing Operations | umoConfiguration | Scheduler | intraDay. 327
Marketing Operations | umoConfiguration | Scheduler | daily 328
Marketing Operations | umoConfiguration | Notifications. 328
Marketing Operations | umoConfiguration | Notifications | Email. 330
Marketing Operations | umoConfiguration | Notifications | project 332
Marketing Operations | umoConfiguration | Notifications | projectRequest. 334

Marketing Operations | umoConfiguration | Notifications | program 334
Marketing Operations | umoConfiguration | Notifications | marketingObject 335
Marketing Operations | umoConfiguration | Notifications | approval 335
Marketing Operations | umoConfiguration | Notifications | asset 336
Marketing Operations | umoConfiguration | Notifications | invoice 337
Campaign configuration properties 338
Campaign 338
Campaign | collaborate 339
Campaign | navigation 339
Campaign | caching 343
Campaign | partitions 345
Campaign | partitions | partition[n] | eMessage 345
Campaign | partitions | partition[n] | eMessage | contactAndResponseHistTracking . 347
Campaign | partitions | partition[n] | Engage 349
Campaign | partitions | partition[n] | Engage | contactAndResponseHistTracking 353
Campaign | partitions | partition[n] | UBX . . 354
Campaign | partitions | partition[n] | UBX | Event Download Schedule 355
Campaign | partitions | partition[n] | Coremetrics 355
Campaign | partitions | partition[n] | reports 357
Campaign | partitions | partition[n] | validation 359
Campaign | partitions | partition[n] | audienceLevels | audienceLevel 360
Campaign | partitions | partition[n] | audienceLevels | audienceLevel | field[n]. . . 361
Campaign | partitions | partition[n] | dataSources 361
Campaign | partitions | partition[n] | systemTableMapping 422
Campaign | partitions | partition[n] | server 422
Campaign | partitions | partition[n] | server | systemCodes 423
Campaign | partitions | partition[n] | server | encoding 426
Campaign | partitions | partition[n] | server | timeout 427
Campaign | partitions | partition[n] | server | collaborate 428
Campaign | partitions | partition[n] | server | spss 429
Campaign | partitions | partition[n] | server | permissions 430
Campaign | partitions | partition[n] | server | flowchartConfig 432
Campaign | partitions | partition[n] | server | flowchartSave 434
Campaign | partitions | partition[n] | server | dataProcessing 435
Campaign | partitions | partition[n] | server | optimization. 440
Campaign | partitions | partition[n] | server | logging 443
Campaign | partitions | partition[n] | server | flowchartRun 448
Campaign | partitions | partition[n] | server | profile 449
Campaign | partitions | partition[n] | server | internal 450
Campaign | partitions | partition[n] | server | fileDialog. 453
Campaign | partitions | partition[n] | offerCodeGenerator 454
Campaign | monitoring 455
Campaign | ProductReindex 457
Campaign | unicaACListener 457
Campaign | unicaACListener | node [n] . . . 465
Campaign | campaignClustering 467
Campaign | unicaACOOptAdmin 470
Campaign | server 471
Campaign | logging 472
Campaign | proxy 472
eMessage configuration properties 473
eMessage | serverComponentsAndLocations | hostedServices 473
eMessage | partitions | partition[n] | hostedAccountInfo 474
eMessage | partitions | partition[n] | dataSources | systemTables 474
eMessage | partitions | partition[n] | recipientListUploader. 478
eMessage | partitions | partition[n] | responseContactTracker 478
Interact configuration properties 480
Interact runtime environment configuration properties 480
Interact design environment configuration properties 548
Contact Optimization configuration properties . . 569
Campaign | unicaACOListener 569
Campaign | unicaACOOptAdmin 571
Campaign | partitions | partition[n] | Optimize | sessionRunMonitor. 573
Campaign | partitions | partition[n] | Optimize | MemoryTuning 573
Campaign | partitions | partition[n] | Optimize | AgentTemplateTables 573
Campaign | partitions | partition[n] | Optimize | userTemplateTables. 574
Campaign | partitions | partition[n] | Optimize | TestRun 574
Campaign | partitions | partition[n] | Optimize | AlgorithmTuning 574
Campaign | partitions | partition[n] | Optimize | Debug 579
Campaign | partitions | partition[n] | Optimize | DatabaseTuning 579
Campaign | partitions | partition[n] | Optimize | logging. 580
Distributed Marketing configuration properties . . 583
Collaborate | navigation 583
Collaborate | UDM Configuration Settings . . 584

Collaborate | UDM Configuration Settings | Attachment 586
Collaborate | UDM Configuration Settings | Attachment Folders 592
Collaborate | UDM Configuration Settings | Campaign Integration 593
Collaborate | UDM Configuration Settings | Datasource 593
Collaborate | UDM Configuration Settings | Flowchart 594
Collaborate | UDM Configuration Settings | History 597
Collaborate | UDM Configuration Settings | Integration Services 597
Collaborate | UDM Configuration Settings | Listing Pages 598
Collaborate | UDM Configuration Settings | List Manager 599
Collaborate | UDM Configuration Settings | Lookup Cleanup 601
Collaborate | UDM Configuration Settings | Notifications. 601
Collaborate | UDM Configuration Settings | Performance. 607
Collaborate | UDM Configuration Settings | Read Only Lookup Tables 607
Collaborate | UDM Configuration Settings | Reports 608
Collaborate | UDM Configuration Settings | Siblings 609
Collaborate | UDM Configuration Settings | Templates 609
Collaborate | UDM Configuration Settings | Workflow. 610
IBM SPSS Modeler Advantage Enterprise Marketing Management Edition configuration properties 612
SPSS | integration. 612
SPSS | integration | partitions | partition [n] 613
SPSS | navigation. 613
Opportunity Detect and Interact Advanced Patterns configuration properties 614
IBM Opportunity Detect and Interact Advanced Patterns | Navigation 614
IBM Opportunity Detect and Interact Advanced Patterns | System | Streams Remote Control Web Service 616
IBM Opportunity Detect and Interact Advanced Patterns | System | Real Time Connector . . . 616
IBM Opportunity Detect and Interact Advanced Patterns | System | Monitoring 616
IBM Opportunity Detect and Interact Advanced Patterns | System | Processing Options . . . 617
IBM Opportunity Detect and Interact Advanced Patterns | logging. 617
IBM Interact Advanced Patterns | System | Interact Design Service 618

Chapter 22. Customization of stylesheets and images in the IBM Marketing Software user interface . . 619
Preparing your corporate theme 619
Applying your corporate theme 620

Before you contact IBM technical support 621

Notices 623
Trademarks 625
Privacy Policy and Terms of Use Considerations 625

Chapter 1. Introduction to IBM Marketing Platform

IBM® Marketing Platform provides security, configuration, notification, and dashboard features for IBM Marketing Software products.

Marketing Platform provides a common user interface for IBM Marketing Software products, as well as the infrastructure for the following features.

- Support for reporting in many products in IBM Marketing Software.
- Support for security in IBM applications, including authentication and authorization.
- Configuration management, including setting user locale preferences and an interface for editing configuration properties for some IBM Marketing Software applications.
- A scheduler that enables you to configure a process to run at intervals that you define.
- Dashboard pages that you can configure to include information useful to groups of users who fill various roles within your organization.
- Support and the user interface for alerts and notifications.
- Security audit reports.

About Marketing Platform security features

The security features in Marketing Platform consist of a central repository and web-based interface where IBM Marketing Software internal users are defined and where users are assigned various levels of access to functions within IBM Marketing Software applications.

IBM Marketing Software applications use the security features of Marketing Platform to authenticate users, check user application access rights, and store user database credentials and other necessary credentials.

Security technologies used in Marketing Platform

Marketing Platform employs industry-standard encryption methods to perform authentication and enforce security across all IBM Marketing Software applications. User and database passwords are protected using a variety of encryption technologies.

Permission management through roles

Marketing Platform defines the user's basic access to the functions within most IBM Marketing Software applications. In addition, for Campaign and Marketing Platform, you can control a user's access to functions and objects within the application.

You can assign various permissions to roles. You can then manage user permissions in either of the following ways.

- By assigning roles to individual users
- By assigning roles to groups and then making users a member of that group

© Copyright IBM Corp. 1999, 2018

About Campaign partitions

Marketing Platform provides support for partitions in the Campaign family of products. Partitions provide a way to secure the data associated with different groups of users. When you configure Campaign or a related IBM Marketing Software application to operate with multiple partitions, each partition appears to application users as a separate instance of the application, with no indication that other partitions exist on the same system.

About groups

A subgroup inherits the roles assigned to its parents. An administrator can define an unlimited number of groups, and any user can be a member of multiple groups. This makes it easy to create different combinations of roles. For example, a user could be an eMessage administrator and a Campaign user with no administration privileges.

A group can belong to only one partition.

Data source credential management

Both users and administrators can set up the user's data source credentials in advance, so the user is not prompted to provide data source credentials when working with an IBM application that requires access to a data source.

Integration with external user and group management systems

Marketing Platform can be configured to integrate with external systems that are used to manage users and resources centrally. These include Windows Active Directory Server, other supported LDAP directory servers, and web access control platforms such as Netegrity SiteMinder and IBM Security Access Manager. This reduces errors, support costs, and the time needed to deploy an application in production.

SAML 2.0 support

Marketing Platform supports SAML (Security Assertion Markup Language) 2.0 for the following.

- SAML 2.0 federated authentication, which enables single sign-on access among diverse applications.

  You can use federated authentication to implement single sign-on between IBM Marketing Software applications and other IBM applications or third-party applications.

  The Marketing Platform installation includes the following components that support federated authentication.

  – An identity provider server WAR file.

  – A client JAR file that you can use with Java™ applications to generate and parse SAML 2.0 assertions. The Java products that you integrate with IBM Marketing Software use the assertions to communicate with the identity provider server.

- SAML 2.0 single sign-on

  A fully functional SAML 2.0 IdP server is a prerequisite for this integration.

After you set up the required configuration properties and a metadata file, users who attempt to log in through the Marketing Platform login page are authenticated through your organization's SAML 2.0 Identity Provider (IdP) server.

Users who are logged in to any application that uses the IdP server for authentication can access IBM Marketing SoftwareIBM Marketing Software without logging in again.

Data filters

Marketing Platform supports configurable data filters that allow you to specify data access restrictions in IBM Marketing Software products. Data filters make it possible to restrict the customer data that an IBM Marketing Software user can view and work with in IBM applications.

Configuration management

The Configuration page provides access to the central configuration properties for IBM Marketing Software applications.

Users with Admin privileges in the Marketing Platform can use the Configuration page to do the following.

- Browse configuration properties, which are organized by product into a hierarchy of categories and sub-categories.
- Edit the values of configuration properties.
- Delete some categories (categories that you can delete display a **Delete Category** link on the Settings page).

You can make additional changes on the Configuration page using the configTool utility provided with Marketing Platform.

Related reference:

"configTool" on page 222

Localization in IBM Marketing Software

Marketing Platform supports localization through its character set encoding and by enabling an administrator to set locale preferences for individual users or all users. Users can also set their own locale preferences.

For both internal and external users, you can set locale preferences on a per-user basis or across the IBM applications that support this feature. This preference setting affects the display of language, time, numbers, and dates in IBM Marketing Software applications.

Marketing Platform supports UTF-8 as the default character set encoding, which allows users to enter data in any language (for example Chinese or Japanese). However, note that full support for any character set in Marketing Platform also depends on the configuration of the following:

- Marketing Platform system table database
- The client machines and browsers used to access IBM Marketing Software.

The common user interface

Marketing Platform provides a common access point and user interface for IBM Marketing Software applications.

The common interface provides the following features.

- When multiple IBM Marketing Software products are installed, you can navigate between products without launching new windows.
- You can view a listing of the pages that you have recently visited, and navigate back to any of those pages using the **Recent** menu.
- You can set an IBM Marketing Software page as a home page (the first page you see when you log in) and you can return to that page at any time by clicking the Home icon.
- You can access the search function for each installed product using the **Search** field. The context of this search function is the page you are viewing. For example, if you are viewing a list of campaigns within Campaign, a search would take place across campaigns. If you wanted to search for a Marketing Operations project, you would perform the search while viewing a list of Marketing Operations projects.

Logging in to IBM Marketing Software

Use this procedure to log in to IBM Marketing Software.

You need the following.

- An intranet (network) connection to access your IBM Marketing Software server.
- A supported browser installed on your computer.
- User name and password to sign in to IBM Marketing Software.
- The URL to access IBM Marketing Software on your network.

The URL is:

`http://`*`host.domain.com:port`*`/unica`

where

`host` is the machine where Marketing Platform is installed.

`domain.com` is the domain in which the host machine resides.

`port` is the port number where the Marketing Platform application server is listening.

Note: The following procedure assumes that you are logging in with an account that has Admin access to Marketing Platform.

Access the IBM Marketing Software URL using your browser.

- If IBM Marketing Software is configured to integrate with Windows Active Directory or with a web access control platform, and you are logged in to that system, you see the default dashboard page. Your login is complete.

- If you see the login screen, log in using the default administrator credentials. In a single-partition environment, use `asm_admin` with `password` as the password. In a multi-partition environment, use `platform_admin` with `password` as the password.

  A prompt asks you to change the password. You can enter the existing password, but for good security you should choose a new one.
- If IBM Marketing Software is configured to use SSL, you may be prompted to accept a digital security certificate the first time you sign in. Click **Yes** to accept the certificate.

If your login is successful, IBM Marketing Software displays the default dashboard page.

With the default permissions assigned to Marketing Platform administrator accounts, you can administer user accounts and security using the options listed under the **Settings** menu. To perform the highest level administration tasks for IBM Marketing Software dashboards, you must log in as **platform_admin**.

Marketing Platform documentation and help

IBM Marketing Platform provides documentation and help for users, administrators, and developers.

Table 1. Get up and running

Task	Documentation
View a list of new features, known issues, and workarounds	*IBM Marketing Platform Release Notes*
Learn about the structure of the Marketing Platform database	*IBM Marketing Platform System Tables*
Install or upgrade Marketing Platform and deploy the Marketing Platform web application	One of the following guides: • *IBM Marketing Platform Installation Guide* • *IBM Marketing Platform Upgrade Guide*
Implement the IBM Cognos® reports provided with IBM Marketing Software	*IBM Marketing Software Reports Installation and Configuration Guide*

Table 2. Configure and use Marketing Platform

Task	Documentation
• Adjust configuration and security settings for IBM products • Integrate with external systems such as LDAP and web access control • Implement single sign-on with diverse applications using SAML 2.0-based federated authentication or single sign-on • Run utilities to perform maintenance on IBM products • Configure and use audit event tracking • Schedule runs of IBM Marketing Software objects	*IBM Marketing Platform Administrator's Guide*

Table 3. Get help

Task	Instructions
Open online help	1. To open a context-sensitive help topic, choose **Help > Help for this page**. 2. To view complete online help, choose **Help > Product documentation** and click the **IBM Marketing Platform in Knowledge Center** link. You must have web access to view context-sensitive online help. Contact IBM Support for more information about accessing and installing IBM Knowledge Center locally for offline documentation.
Obtain PDFs	Use either of the following methods: • Choose **Help > Product documentation** to access Marketing Platform PDFs and help. • Choose **Help > All IBM Marketing Software Documentation** to access documentation for all products.
IBM Knowledge Center	To access the IBM Knowledge Center, choose **Help > Support for this product**.
Get support	Go to http://www.ibm.com/support to access the IBM Support Portal.

Chapter 2. IBM Marketing Software user account management

You can manage the attributes of user accounts created using IBM Marketing Platform user interface, which we refer to as internal accounts. This is in contrast to external user accounts, which are imported from an external system such as an LDAP server or web access control system.

External accounts are managed in the external system.

Types of user accounts: internal and external

When IBM Marketing Software is integrated with an external server (such as a supported LDAP server or a web access control system), it supports two types of user accounts: internal and external.

- **Internal** - User accounts that are created within IBM Marketing Software using the security user interface. These users are authenticated through IBM Marketing Software.
- **External** - User accounts that are imported into IBM Marketing Software through synchronization with an external server. This synchronization occurs only if IBM Marketing Software has been configured to integrate with the external server. These users are authenticated through the external server. Examples of external servers are LDAP and web access control servers.

Depending on your configuration, you might have only internal users, only external users, or a combination of both. If you integrate IBM Marketing Software with Windows Active Directory and enable LDAP, you can have only external users.

For more information about integrating IBM Marketing Software with an LDAP or Windows Active Directory server, see the relevant sections in this guide.

Management of external users

Usually, the attributes of external user accounts are managed through the external system. Within IBM Marketing Software, you can control the following aspects of an external user account: data sources, notification preferences, locale preference for IBM Marketing Software applications, and membership in internal groups (but not external groups).

Identifying internal and external users in the IBM Marketing Software interface

In the Users section of IBM Marketing Software, internal and internal users have different icons, as follows.

- Internal -
- External -

© Copyright IBM Corp. 1999, 2018

Properties of internal user accounts

Administrators can manage the properties of user accounts that have been created using the Marketing Platform user interface.

When a user forgets a password

Marketing Platform stores internal user passwords in hashed form, and these stored passwords cannot be restored to clear text. You must assign a new password for users with an internal account who forget their password.

Resetting a password

Users with internal accounts can change their own passwords by providing the original password and entering and confirming the new password. The IBM Marketing Software administrator can also reset any user password as needed.

Password expiration dates

You can set password expiration intervals for all users on the Configuration page. You can also set expiration dates on a per-user basis for users (when the system-wide expiration date is not set to never expire).

System status of user accounts

The system status of a user is either active or disabled. A user with a disabled account cannot log in to any IBM Marketing Software application. If a disabled user account was formerly active, with membership in one or more groups, you can make the account active again. When you make a disabled user account active the group memberships are retained.

Alternate login

You can specify an alternate login for any user account. An alternate login is typically required when the Campaign listener runs as root on a UNIX-type system.

Data sources

A user needs appropriate credentials to access the data sources used by some IBM Marketing Software applications. You can enter these credentials as a data source in the user account properties.

When a user is working in an IBM Marketing Software application such as Campaign and is prompted for data source information, the IBM Marketing Software application stores this information in Marketing Platform data store. These data sources appear in the data source list for the user in Marketing Platform even though they were not created using the IBM Marketing Software interface.

Adding internal user accounts

Use this procedure to add internal user accounts.

1. Click **Settings > Users**.
2. Click **New user**.
3. Complete the form and click **Save changes**.

   Use caution if you employ special characters in login names. Allowed special characters are listed in the New user page reference.
4. Click **OK**.

   The new user name appears in the list.

Related reference:

"The user management pages" on page 12

Deleting internal user accounts

Use this procedure to delete internal user accounts.

Important: If Campaign permissions are set up in a way that restricts ownership or access to a Campaign object to a single user, deleting the account of that user makes the object inaccessible. Instead, you should disable rather than delete such accounts.

1. Click **Settings > Users**.
2. Click the user name of the account you want to delete.
3. Click **OK**.

Related reference:

"The user management pages" on page 12

Changing internal user password expiration dates

Use this procedure to change password expiration dates for internal users.

Restriction: If the system-wide password expiration property **General | Password settings | Validity (in days)** is set to zero, you cannot change the password expiration date of any internal user.

1. Click **Settings > Users**.
2. Click the user name.
3. Click the **Edit properties** link at the bottom of the page.
4. Change the date in the **Password expiration** field.
5. Click **OK**.

Related reference:

"The user management pages" on page 12

Resetting internal user passwords

Use this procedure to reset internal user passwords.

1. Click **Settings > Users**.

   The **Users** list is displayed in the left pane.
2. Click the user name you want to change.
3. Click the **Reset password** link at the bottom of the page.
4. Enter the new password in the **Password** field.
5. Enter the same password in the **Confirm** field.
6. Click **Save changes** to save your changes.
7. Click **OK**.

   Note: When user passwords are reset, users are prompted to change their password the next time they log in to an IBM Marketing Software application.

Related reference:

"The user management pages" on page 12

Changing internal user account properties

Use this procedure to change the properties of internal user account.

1. Click **Settings > Users**.
2. Click the name of the account you want to change.
3. Click the **Edit properties** link at the bottom of the page.
4. Edit the fields as needed.
5. Click **Save changes** to save your changes.
6. Click **OK**.

Related tasks:

"Resetting internal user passwords"

Related reference:

"The user management pages" on page 12

Changing internal user system status

Use this procedure to change the system status of internal users.

1. Click **Settings > Users**.
2. Click the name of the account you want to change.
3. Click the **Edit properties** link at the bottom of the page.
4. Select the status in the **Status** drop-down list. The options are **ACTIVE** and **DISABLED**.

   Note: If you select **DISABLED**, the user will no longer be able to log in to any IBM Marketing Software applications. Users with Admin access to Marketing Platform cannot disable themselves.
5. Click **Save changes** to save your changes.
6. Click **OK**.

Related reference:

"The user management pages" on page 12

Adding internal user data sources

Use this procedure to add data sources for internal users.

1. Click **Settings > Users**.
2. Click the name of the account you want to change.
3. Click the **Edit data sources** link at the bottom of the page.
4. Click **Add new**.
5. Complete the form and click **Save changes** to save your changes.
6. Click **OK**.

Related reference:

"The user management pages" on page 12

Changing internal user data sources

Use this procedure to change data source passwords or login names.

1. Click **Settings > Users**.
2. Click the name of the account you want to change.
3. Click the **Edit data sources** link at the bottom of the page.
4. Click the **Data source name** you want to change.
5. Edit the fields.

   If you do not set a new password, the old one is retained.
6. Complete the form and click **Save changes** to save your changes.
7. Click **OK**.

Related tasks:

"Changing internal user account properties" on page 10

Related reference:

"The user management pages" on page 12

Deleting internal user data sources

Use this procedure to delete internal user data sources.

1. Click **Settings > Users**.
2. Click the name of the account you want to change.
3. Click the **Edit Data Sources** link at the bottom of the page.
4. Click the name of the data source you want to delete.
5. Click **Delete**.
6. Click **OK**.

Related reference:

"The user management pages" on page 12

The user management pages

Refer to this table if you need help completing the fields on the Users page.

The New user page

Table 4. Fields on the New user page

Field	Description
First name	The user's first name.
Last name	The user's last name.
Login	The user's login name. This is the only required field. Only the following special characters are allowed in login names. • Upper and lower case alphabetic characters (A-Za-z) • Numbers (0-9) • The 'at' sign (@) • Hyphen (-) • Underscore (_) • Dot (.) • Double byte characters (such as Chinese characters) Do not include other special characters (including spaces) in the login name.
Password	A password for the user. Follow these rules when creating a password. • Passwords are case-sensitive. For example, `password`is not the same as `Password`. • You may use any character when you create or reset a password in IBM Marketing Software. Additional password requirements are set on the Configuration page. To see what they are for your installation of IBM Marketing Software, click the **Password Rules**link next to the **Password** field.
Confirm password	The same password you entered in the **Password**field.
Title	The user's title.
Department	The user's department.
Company	The user's company.
Country	The user's country.
Address	The user's address.
Work phone	The user's work phone number.
Mobile phone	The user's mobile phone number.
Home phone	The user's home phone number.
Email address	The user's email address. This field must conform to email addresses as defined in RFC 821. See RFC 821for details.
Alternate login	The user's UNIX login name, if one exists. An alternate login is typically required when the Campaignlistener runs as root on a UNIX-type system.
Status	Select ACTIVE or DISABLED from the drop-down list. ACTIVE is selected by default. Disabled users are prevented from logging in to all IBM Marketing Softwareapplications.

The Edit properties page

The fields are the same as the fields on the New user page, except for the ones shown in the following table.

Table 5. Fields on the Edit properties page

Field	Description
Password	This field is not available on the Edit properties page.
Login	This field is not available on the Edit properties page.
Password expiration	The date in the format appropriate for your locale (for example, for en_US, the format is `MM, dd, yyyy`). You cannot change a user's expiration date when the system-wide expiration date is set to never expire.
IBM Digital Analytics user name	When integration is enabled with IBM Digital Analytics, and you choose to create users manually, you enter the user's Digital Analyticsuser name here as part of the configuration process.

The Reset password page

Table 6. Fields on the Reset password page

Field	Description
Password	The new password.
Confirm	The same password you entered in **Password** field.

The New Data Source and Edit Data Source Properties pages

Table 7. Fields on the Data Source pages

Field	Description
Data source	The name of a data source you want the user to be able to access from an IBM Marketing Softwareapplication. IBM Marketing Softwarenames preserve case for display purposes, but use case-insensitive rules for comparison and creation (for example, you cannot create both `customer`and `Customer`data source names).
Data source login	The login name for this data source.
Data source password	The password for this data source. You can leave this field empty, if the data source account does not have a password.
Confirm password	The password again (leave empty if you left the **Data Source Password**field empty).

Locale preference

You can set the locale for both internal and external users. This setting affects the display of language, time, numbers, and dates in IBM Marketing Software applications.

There are two ways to set locale in Marketing Platform.

Globally
A configuration property, `Platform | Region setting`, on the **Settings > Configuration** page, sets the locale globally.

Per user

An attribute on the **Settings > Users** page sets the locale for individual users. This setting overrides the global setting.

Availability of locales that you can set either per user or globally may vary depending on the IBM Marketing Software application, and not all IBM Marketing Software applications support this locale setting in Marketing Platform. See specific product documentation to determine availability and support for the `Region setting` property.

Note: Availability of locales that you can set either per user or globally may vary depending on the IBM Marketing Software application. Not all IBM Marketing Software applications support this locale setting. See specific product documentation to determine availability and support for the locale settings in IBM Marketing Software.

Related concepts:

"The default user locale preference" on page 35

Setting the user locale preference

Use this procedure to set the locale preference for a user.

1. Click **Settings > Users**.
2. Click the user name you for which you want to set locale preferences.
3. Click the **Edit preferences** link at the bottom of the page.
4. Click **IBM Marketing Platform** in the left pane.
5. Select an option from the **Region**drop-down list.
6. Click **Save changes**.

Synchronization of external users

When IBM Marketing Software is configured to integrate with a Windows Active Directory or LDAP server, users and groups are automatically synchronized automatically at pre-defined intervals.

Automatic synchronization has limited functionality.

- Automatic synchronization updates user attributes only. Because group membership changes such as adding, removing, or changing members in a group require administrator oversight, import of these changes is confined to the manual synchronization process by default.
- Users deleted from the LDAP server are not deleted during automatic synchronization.

You can force a full synchronization of all users and groups by using the Synchronize function in the Users area of IBM Marketing Software.

Forcing synchronization of external users

Use this procedure to force synchronization of users when IBM Marketing Software is integrated with an LDAP server or web access control system.

1. Log in to IBM Marketing Software and click **Settings > Users**.
2. Click **Synchronize**.

   Users and groups are synchronized.

Chapter 3. Security management

IBM Marketing Platform supports roles and permissions to control user access to objects and features in IBM Marketing Software applications.

For the most part, only Marketing Platform itself and Campaign use the User roles and permissions page to manage users' application access in detail.

The other IBM Marketing Software products use some basic application access roles set on the User roles and permissions page, and either do not have detailed security settings, or the settings are not managed on the User roles and permissions page.

For example, in Marketing Operations, setting up the basic roles on the User roles and permissions page is only the starting point for developing a customized security scheme. Marketing Operations has a detailed security scheme you can manage through a user interface on the Marketing Operations pages.

This guide explains how to use the functions on the User roles and permissions page, and describes the basic security roles and persmissions shown on this page for the various products. For products other than Marketing Platform, if you do not see the security management information you need in this guide, see the product's documentation.

Permissions and tasks of the security administrator in Marketing Platform

Only users with either the AdminRole or PlatformAdminRole role in Marketing Platform have access to security administration features for user accounts other than their own.

In a multi-partition environment, only users with the PlatformAdminRole role can administer users across partitions. Users with the AdminRole role can administer users in their own partition only.

The security administrator performs the following tasks on the User groups and User roles & permissions pages.

- Create internal groups and manage their memberships and partition assignments.
- Create roles for Marketing Platform and Campaign, if necessary, and assign permissions to these roles.
- Manage user access to IBM Marketing Software applications by assigning roles to individual users and/or to internal and external groups.

Read this overview to gain an understanding of the following.

- The difference between internal and external groups
- The process of creating internal groups and assigning roles and permissions
- The properties of internal groups
- The pre-configured user accounts, groups, and roles in Marketing Platform

© Copyright IBM Corp. 1999, 2018

Special characters in role and policy names

You may use only the following characters when you create role and policy names.

- Upper and lower case alphabetic characters (A-Z)
- Numbers (0-9)
- Single quote (')
- Hyphen (-)
- Underscore (_)
- The 'at' sign (@)
- Forward slash (/)
- Parenthesis
- Colon (:)
- Semi-colon (;)
- Space (except as the first character)
- Double byte characters (such as Chinese characters)

Roles and permissions in Marketing Platform and Campaign

Roles in Marketing Platform and Campaign are a configurable collection of permissions. For each role in Marketing Platform and Campaign, you can specify permissions that control access to the application.

You can use the default roles or create new roles. The set of available permissions is defined by the system; you cannot create a new permission.

About role assignment

Generally, you should give users roles with permissions that reflect the functions that users perform in your organization when they use IBM Marketing Software. You can assign roles to a group or to an individual user. The advantage of assigning roles by group is that you can assign a combination of roles to the group, and if you later want to change that combination, you can do it in one place rather than having to do it multiple times for multiple users. When you assign roles by group, you add and remove users from your groups to control user access.

How the system evaluates roles

If a user has multiple roles, the system evaluates permissions from all those roles together. The ability to perform a function on a particular object is then granted or denied based on the aggregated permissions from all roles. In the case of Campaign, the ability to perform a function on a particular object is granted or denied based on the security policy of the object.

Overview of managing user application access in Marketing Platform

Using Marketing Platform security administration features to manage user application access is a multi-step process. The following procedure provides an overview of the basic process, which is described in detail in the remainder of this guide.

1. Plan the roles you want to use to control user access to IBM Marketing Software products. Configure roles and their permissions as needed.

2. Plan what groups you need to fulfill your security requirements. You may have only internal groups, only external groups, or a combination of both, depending on how your system is configured.
3. Create any necessary internal and external groups.
4. Assign your groups to roles.
5. If you have only internal user accounts, create any internal user accounts as needed.
6. Assign users to groups, or assign roles to individual users, based on the application access you want the users to have.

Related concepts:

"Types of groups: internal and external"

"Partitions and security management" on page 18

"Pre-configured users and roles" on page 19

"Cross-partition administration privileges" on page 20

Types of groups: internal and external

When IBM Marketing Software is integrated with an external server (such as a supported LDAP server or a web access control system), it supports two types of groups: internal and external.

- **Internal** - Groups that are created within IBM Marketing Software using the security user interface. These users are authenticated through IBM Marketing Software.
- **External** - IBM Marketing Software groups that are mapped to groups in the external system. Examples of eternal servers are LDAP and web access control servers.

  Attention: A group referred to as an external group in this guide is one that is actually created in IBM Marketing Software but is mapped to an external system.

Depending on your configuration, you may have only internal groups, only external groups, or a combination of both.

For more information about integrating IBM Marketing Software with an LDAP or Windows Active Directory server, see the relevant sections of this guide.

Management of external groups

The membership of external groups is managed in the external system.

You can assign roles to IBM Marketing Software external groups just as you do to internal groups.

Management of internal groups and subgroups

You can define an unlimited number of internal groups, and any internal or external user can be a member of multiple internal groups and subgroups.

A subgroup does not inherit the user members assigned to its parents, but it does inherit the roles assigned to its parents. A group and its subgroups always belong to one partition.

Only internal groups may be assigned to a partition, and only the platform_admin user, or another account with the PlatformAdminRole role, can create groups in all partitions in a multi-partition environment.

Related tasks:

"Overview of managing user application access in Marketing Platform" on page 16

Partitions and security management

Partitions in Campaign and related products provide a way to secure the data associated with different groups of users. With partitioning, a user's partition appears as if it were a separate running instance of Campaign, with no indication that other partitions are running on the same system. This section describes special security management considerations in a multi-partition environment.

User membership in a partition

You assign users to a partition based on their group membership. You assign a group to a partition and then assign users to a group to give them access to a partition.

A group or subgroup may be assigned to just one partition, and parent groups do not acquire the partition assignments of their subgroups. Only the platform_admin user, or another account with the PlatformAdminRole role, can assign a group to a partition.

You should make a user a member of only one partition.

About roles and partitions

A role always exists in the context of a partition. In a single-partition environment, all roles are automatically created within the default partition, partition1. In a multi-partition environment, a role is created in the partition of the user who created it. The exception is the platform_admin user and any other accounts with the PlatformAdminRole role; these accounts can create roles in any partition.

More information about partitions

This section provides instructions on assigning a group to a partition, and assigning users to groups. For complete details on configuring partitions, see the Campaign installation documentation.

Related tasks:

"Overview of managing user application access in Marketing Platform" on page 16

Pre-configured users and roles

When IBM Marketing Software is first installed, three users are pre-configured and are assigned system-defined roles in Marketing Platform and Campaign, as described in this section.

These internal user accounts all have "password" as the default password.

The platform_admin user account

The platform_admin user account is designed to allow an IBM Marketing Software administrator to manage product configuration, users, and groups across all partitions in a multi-partition environment, and to use all Marketing Platform features (except reporting, which has its own roles) without any filtering by partition. By default, this account has the following roles in Marketing Platform.

- In Marketing Platform, in the default partition, partition1
  - AdminRole
  - UserRole
  - PlatformAdminRole

  These roles allow the platform_admin user to perform all administrative tasks within Marketing Platform, except for the reporting functions. When additional partitions are created, the platform_admin user can access and administer users, groups, roles, and configuration within the additional partitions.

  The PlatformAdminRole role is unique in that no user can modify permissions for this role, and only a user with this role can assign the PlatformAdminRole role to another user.
- In Campaign, in the default partition, partition1
  - The Global policy Admin role

  This role allows the platform_admin user to perform all tasks within Campaign.

By default, this user does not have access to any IBM Marketing Software products beyond Marketing Platform and Campaign.

The asm_admin user account

The asm_admin user account is designed to allow an IBM Marketing Software administrator to manage users and groups in a single-partition environment, and to use all Marketing Platform features (except reporting, which has its own roles). This account has the following roles.

- In Marketing Platform, in the default partition, partition1
  - AdminRole
  - UserRole

  With the exceptions noted below, these roles allow the asm_admin user to perform all administrative tasks within Marketing Platform within the partition to which asm_admin belongs, which is partition1 by default.

  These roles allow this user to administer the Configuration page, which does not filter by partition for any user. For this reason, you should remove the Administer Configuration page permission from the AdminRole role in Marketing Platform, and reserve configuration tasks for the platform_admin user.

  The exceptions are as follows.
  - To access reporting functions, you must grant the Reports System role.

– This user cannot assign the PlatformAdminRole role to any user or group.

The demo account

The demo account has the following roles.

- In Marketing Platform, in the default partition, partition1
  – UserRole

  This role allows the demo user to view and modify his or her own account attributes on the Users page, but not to change roles or partitions for his or her own account or access any of the other features contained within Marketing Platform. By default, this user does not have access to any of the IBM Marketing Software products.
- In Campaign, in the default partition, partition1
  – The Global policy Review role

  This role allows the demo user to create bookmarks and to view campaigns, sessions, offers, segments, and reporting in Campaign.

Related tasks:

"Overview of managing user application access in Marketing Platform" on page 16

Cross-partition administration privileges

In a multi-partition environment, at least one user account with the PlatformAdminRole role in Marketing Platform is required, to enable you to administer security for IBM Marketing Software users across all partitions.

The platform_admin account is pre-configured with the PlatformAdminRole role. The platform_admin account is a superuser account that cannot be deleted or disabled through the Users functions in IBM Marketing Software. However, this account is subject to the password constraints of any other user. For example, someone attempting to log in as platform_admin might enter an incorrect password N times in a row. Depending on the password rules in effect, the platform_admin account might be disabled in the system. To restore this account, you must take one of the following actions.

- If you have another user with the PlatformAdminRole role in Marketing Platform, log in as that user and reset the platform_admin user's password or create another account with the PlatformAdminRole role in Marketing Platform.
- If you have only one user with the PlatformAdminRole role in Marketing Platform (for example, platform_admin), and this user is disabled, you can create a new platform_admin account using the `restoreAccess` utility provided with Marketing Platform.

To avoid a situation where you must restore PlatformAdminRole access using the `restoreAccess` utility, it is a good practice to create more than one account with PlatformAdminRole privileges.

Related tasks:

"Overview of managing user application access in Marketing Platform" on page 16

Related reference:

"restoreAccess" on page 231

Adding an internal group

Use this procedure to add an internal group.

1. Click **Settings > User groups**.
2. Click **New group** above the **Group Hierarchy** list.
3. Complete the **Group name** and **Description** fields.

   Important: Do not give the group a the same name as system-defined roles. For example, do not name a group "Admin," which is a role name used in Campaign. Doing so can cause problems during upgrades.
4. Click **Save changes**.

   The new group's name appears in the **Group hierarchy** list.

Related reference:

"The user group management pages" on page 23

Adding a subgroup

Use this procedure to add an internal subgroup.

1. Click **Settings > User groups**.
2. Click the name of the group to which you want to add a subgroup.
3. Click **New subgroup**.
4. Complete the **Group name** and **Description** fields.

   Important: Do not give the subgroup a the same name as system-defined roles. For example, do not name a subgroup "Admin," which is a role name used in Campaign. Doing so can cause problems during upgrades.
5. Click **Save changes**.

   The new subgroup is added under the appropriate group in the **Group Hierarchy** list.

   Tip: If the parent group's folder icon is closed, click the plus sign (+) to expand the list.

Related reference:

"The user group management pages" on page 23

Deleting a group or subgroup

Remember, when you delete a group or subgroup, members of the group lose the roles assigned to that group, and any parents of that group also lose those role assignments, unless the roles are also explicitly assigned to the parents.

1. Click **Settings > User groups**.
2. Click the name of the group or subgroup that you want to delete.

   Note: To select a subgroup when the parent group's folder icon is closed, click the plus sign (+) to expand the list.

3. Click the **Delete group** button at the top of the right pane.
4. Click **OK**.

Related reference:

"The user group management pages" on page 23

Changing a group or subgroup description

Use this procedure to change a group or subgroup description.

1. Click **Settings > User groups**.
2. Click the name of the group or subgroup whose description you want to change.

   Note: To select a subgroup when the parent group's folder icon is closed, click the plus sign (+) to expand the list.
3. Click **Edit properties**.
4. Edit the description as desired.
5. Click **Save changes** to save your changes.
6. Click **OK**.

Related reference:

"The user group management pages" on page 23

Assigning a group to a partition

This procedure is necessary only if multiple partitions are configured for Campaign. Only an account with the PlatformAdminRole role, such as the platform_admin user, can perform this task.

1. Determine which groups you want to assign to each partition. Create the groups, if necessary.
2. Click **Settings > User groups**.
3. Click the name of the group or subgroup that you want to assign to a partition.
4. Click **Edit properties**.
5. Select the desired partition from the **Partition ID** drop-down list.

   This field is available only when multiple partitions are configured.
6. Click **Save changes** to save your changes.
7. Click **OK**.

Related reference:

"The user group management pages" on page 23

Adding a user to a group or subgroup

Use this procedure to add a user to a group or subgroup.

1. Click **Settings > Users**.

Note: You can perform the same task on the **User groups** page by clicking the group name and then clicking **Edit Users**.

2. Click the user name you want to change.
3. Click the **Edit groups** link at the bottom of the page.
4. Click a group name in the **Available groups** box to select it.
5. Click the **Add** button.

   The group name moves to the **Groups** box.
6. Click **Save changes** to save your changes.
7. Click **OK**.

   The user account details are displayed, with the group or subgroup you assigned listed.

Related reference:

"The user group management pages"

Removing a user from a group or subgroup

Use this procedure to remove a user from a group or subgroup.

Important: Removing a user from a group or subgroup removes the roles assigned to that group or subgroup from the user.

1. Click **Settings > Users**.
2. Click the user name you want to change.
3. Click the **Edit groups** link at the bottom of the page.
4. Click a group name in the **Groups** box to select it.
5. Click the **Remove** button.

   The group name moves to the **Available groups** box.
6. Click **Save changes** to save your changes.
7. Click **OK**.
8. Click the **Edit properties** link at the bottom of the page.
9. Change the name or description as desired.
10. Click **Save changes** to save your changes.
11. Click **OK**.

Related reference:

"The user group management pages"

The user group management pages

These are the fields you use to configure user groups.

Fields on the New group, New subgroup, and Edit properties pages

Table 8. Fields on the New group, New subgroup, and Edit properties pages

Field	Description
Group name	The group name. The limit is 64 characters. You may use the following characters when you create a group name. • Upper and lower case alphabetic characters (A-Z) • Numbers (0-9) • Single quote (') • Hyphen (-) • Underscore (_) • The 'at' sign (@) • Forward slash (/) • Parenthesis • Colon (:) • Semi-colon (;) • Space (except as the first character) • Double byte characters (such as alpha-numeric Chinese characters) Do not give a group or subgroup a the same name as system-defined roles. For example, do not name a group "Admin," which is a role name used in Campaign. Doing so can cause problems during upgrades. IBM Marketing Software names preserve case for display purposes, but use case-insensitive rules for comparison and creation (for example, you cannot create both Admin and admin as as separate group names). When you create a subgroup, it is a good idea to give your subgroup a name that relates it to its parent group.
Description	The group description. The limit is 256 characters. It is helpful to include the roles you plan to give the group or subgroup in the description. Then you can see at a glance on the group detail page both the roles and users.
Partition ID	Available only when multiple partitions are configured. If you assign a partition to a group, the members of that group are members of that partition. A user can be a member of only one partition.

Fields on the Edit users and Edit roles pages

Table 9. Fields on the Edit users and Edit roles pages

Field	Description
Available groups or Available roles	A list of groups and subgroups or roles to which the user is not assigned.
Groups or Roles	A list of groups and subgroups or roles to which the user is assigned

Creating a role

You should create new roles only for products that have detailed permissions. The reporting function and some IBM Marketing Software products have only basic permissions available, so there is no need to create additional roles for these products.

1. Click **Settings > User roles & permissions**.
2. Click the plus sign next to the product name in the list on the left, and then click the name of the partition where you want to create the role.
3. For Campaign only, if you want to create a new role under the Global Policy, click Global Policy.
4. Click **Add roles and assign permissions**.
5. Click **Add a role**.
6. Enter a name and description for the role.
7. Click **Save changes** to save the role, or **Save and edit permissions** to go to the Permissions page to add or modify permissions for any of the roles in the list.

Related reference:

"Definitions of permission states" on page 27

"Permissions for products that use only basic roles" on page 28

"Permissions for Marketing Platform" on page 29

"Permissions for Opportunity Detect" on page 30

Modifying role permissions

Use this procedure to modify role permissions.

1. Click **Settings > User roles & permissions**.
2. Click the plus sign next to a product in the list on the left, and then click the name of the partition where you want to modify a role.
3. For Campaign only, if you want to create a new role under the Global Policy or a user-created policy, click the policy name.
4. Click **Add roles and assign permissions**.
5. Click **Save and edit permissions**
6. Click the plus sign next to a role group to display all available permissions and the state of those permissions within each role.
7. In the role column where you want to modify permissions, click the box in the permissions rows to set the state to Grant, Deny, or Not granted.
8. Click **Save changes** save your changes.

   You can click **Revert to saved** to undo changes since your last save and remain on the Permissions page, or **Cancel** to discard your changes since your last save and go to the partition or policy page.

Related reference:

"Definitions of permission states" on page 27

"Permissions for products that use only basic roles" on page 28

"Permissions for Marketing Platform" on page 29

"Permissions for Opportunity Detect" on page 30

Removing a role from the system

Use this procedure to remove a role from IBM Marketing Software.

Important: If you remove a role, it is removed from all users and groups to which it was assigned.

1. Click **Settings > User roles & permissions**.
2. Click the plus sign next to a product in the list on the left, and then click the name of the partition where you want to create the role.
3. For Campaign only, if you want to create a new role under the Global Policy, click Global Policy.
4. Click **Add roles and assign permissions**.
5. Click the **Remove** link for the role you want to delete.
6. Click **Save changes**.

Related reference:

"Definitions of permission states" on page 27

"Permissions for products that use only basic roles" on page 28

"Permissions for Marketing Platform" on page 29

"Permissions for Opportunity Detect" on page 30

Assigning a role to or removing a role from a group

If you add a role to a group or remove a role from a group, members of that group acquire or lose that role.

1. Click **Settings > User groups**.
2. Click the name of the group that you want to work with.
3. Click **Assign roles**.

   Roles that are not assigned to the group are shown in the **Available roles** box on the left. Roles that are currently assigned to the group are shown in the **Roles** box on the right.
4. Click a role name in the Available roles box to select it.
5. Click **Add** or **Remove** to move the role name from one box to the other.
6. Click **Save changes** to save your changes.
7. Click **OK**.

Related reference:

"Definitions of permission states" on page 27

"Permissions for products that use only basic roles" on page 28

"Permissions for Marketing Platform" on page 29

"Permissions for Opportunity Detect" on page 30

Assigning a role to or removing a role from a user

Use the Edit roles window to assign a role to or to remove a role from a user.

Complete the following tasks to assign or remove a role from a user:

1. Click **Settings > Users**.
2. Click the name of the user account that you want to work with.
3. Click Edit roles.

   Roles that are not assigned to the user are shown in the **Available Roles** box on the left. Roles that are currently assigned to the user are shown in the **Selected roles** box on the right.
4. Select a role in the **Available roles** box. Complete one of the following tasks:
   - To assign a role to a user, select a role in the **Available roles** box, and click **Add**.
   - To remove a role from a user, select a role in the **Selected roles** box, and click **Remove**.
5. Click **Save changes**, and then click **OK**.

Related reference:

"Definitions of permission states"

"Permissions for products that use only basic roles" on page 28

"Permissions for Marketing Platform" on page 29

"Permissions for Opportunity Detect" on page 30

Definitions of permission states

For each role, you can specify which permissions are granted, not granted, or denied. You set these permissions on the **Settings > User roles and permissions** page.

These states have the following meanings.

- **Granted** - indicated with a check mark . Explicitly grants permission to perform this particular function as long as none of the user's other roles explicitly denies permission.
- **Denied** - indicated with an "X" . Explicitly denies permission to perform this particular function, regardless of any other of the user's roles which might grant permission.
- **Not granted** - indicated with a circle . Does not explicitly grant nor deny permission to perform a particular function. If this permission is not explicitly granted by any of a user's roles, the user is not allowed to perform this function.

Related tasks:

"Creating a role" on page 25

"Modifying role permissions" on page 25

"Removing a role from the system" on page 26

"Assigning a role to or removing a role from a group" on page 26

"Assigning a role to or removing a role from a user"

Permissions for products that use only basic roles

The following table describes the functional definitions of the roles available for the IBM Marketing Software products that use only the basic roles. See the product documentation for additional information.

Table 10. Permissions for products that use only basic roles

IBM Application	Roles
Leads	Leads roles are reserved for future use.
Reports	• ReportsSystem - grants the `report_system` permission, which gives you access to the **Report SQL Generator** and **Sync Report Folder Permissions** options in the **Settings** menu. • ReportsUser - grants the `report_user` permission, which is used by the IBM Authentication Provider installed on the IBM Cognos 11 BI system only. For information about authentication options for the IBM Cognos 11 BI integration and how the IBM Authentication Provider uses the reporting permissions, see the *IBM Marketing Software Reports Installation and Configuration Guide*.
eMessage	• eMessage_Admin - Has full access to all features. • eMessage_User - Reserved for future use. Access is further defined through the security policies in Campaign. See the *IBM eMessage Startup and Administrator's Guide* for details.
Interact	• InteractAdminRole - Has full access to all features.
Distributed Marketing	• collab_admin - Has full access to all features. • corporate - Can use Campaign and Distributed Marketing to develop reusable Lists and On-demand Campaign templates. Can create and execute Corporate Campaigns. • field - Can participate in Corporate Campaigns and can create and execute Lists and On-demand Campaigns in Distributed Marketing.
Marketing Operations	• PlanUserRole - By default, users with the PlanUserRole role have very few permissions enabled in Marketing Operations. They cannot create plans, programs, or projects and have limited access to the Administrative settings. • PlanAdminRole - By default, users with the PlanAdminRole role have most permissions enabled in Marketing Operations, including access to all administrative and configuration settings, allowing a broad range of access. Access is further defined through the security policies in Marketing Operations.

Table 10. Permissions for products that use only basic roles (continued)

IBM Application	Roles
IBM SPSS® Modeler Advantage Enterprise Marketing Management Edition	• SPSSUser - Users with the SPSSUser role can do the following: – Run reports – View items in their Content Repository – Perform scoring • SPSSAdmin - Users with the SPSSAdmin role have all permissions enabled in IBM SPSS Modeler Advantage Enterprise Marketing Management Edition, including access to all administrative and configuration settings.

Related tasks:

"Creating a role" on page 25

"Modifying role permissions" on page 25

"Removing a role from the system" on page 26

"Assigning a role to or removing a role from a group" on page 26

"Assigning a role to or removing a role from a user" on page 27

Permissions for Marketing Platform

The following table describes the permissions you can assign to roles in Marketing Platform.

Table 11. Marketing Platform permissions

Permission	Description
Administer users page	Allows a user to perform all user administration tasks on the Users page for user accounts in his or her own partition: add and delete internal user accounts, and modify attributes, data sources and role assignments
Access users page	Allows a user to view the User page.
Administer User groups page	Allows a user to perform all actions on the User groups page except assign a partition to a group, which can only be done by the platform_admin user. This permission allows a user to create, modify, and delete groups, manage group membership, and assign roles to groups.
Administer User roles page	Allows a user to perform all actions on the User roles & permissions page: create, modify, and delete roles in Marketing Platform and Campaign, and assign users to roles for all listed IBM Marketing Software products.
Administer Configuration page	Allows a user to perform all actions on the Configuration page: modify property values, create new categories from templates, and delete categories that have the **Delete Category** link.
Administer Data filters page	Allows a user to perform all actions on the Data filters page: assign and remove data filter assignments.
Administer Scheduled tasks page	Allows a user to perform all actions on the Schedule management page: view and modify schedule definitions and view runs.
Administer dashboards	Allows a user to perform all actions on the dashboards pages: create, view, modify, and delete dashboards, assign dashboard administrators, and administer dashboard access.

Related tasks:

"Creating a role" on page 25
"Modifying role permissions" on page 25
"Removing a role from the system" on page 26
"Assigning a role to or removing a role from a group" on page 26
"Assigning a role to or removing a role from a user" on page 27

Permissions for Opportunity Detect

The following table describes permissions that you can assign to roles in Opportunity Detect.

All permissions that have the **Not Granted** status are treated as **Denied**.

Table 12. Permissions in Opportunity Detect

Permission	Description
View only	Can access all of the user interface, in view-only mode.
Design triggers	• Can create workspaces and design trigger systems. • Can create, modify, and delete all trigger related resources. • Can access Workspace, Component, Audience Level, Data Source, and Named Value List pages. • Can not access the Server Groups page or the Deployment tab of a workspace. • Can not set off a batch run. • Can not administer objects that the web service creates when Opportunity Detect is integrated with Interact.
Run for testing	• Deploy deployment configurations and run batch deployment configurations on server groups not designated for production. • Can access Server Group page and the Deployment tab of a workspace, but can not designate a server group for production. • Can not deploy deployment configurations or run deployment configurations that use a production server group.
Run for production	• Deploy deployment configurations and run batch deployment configurations on any server group. • Perform all actions on the Server Group page and the Deployment and Batch Run tabs of a workspace, including designating a server group for production.

Table 12. Permissions in Opportunity Detect (continued)

Permission	Description
Administer real time	Manage objects that the web service creates when Opportunity Detect is integrated with Interact to enable real time mode. Allows the following. • Delete workspaces and components created by the web service. • Start and stop real time deployment configurations and update their log level. The user with this permission alone can not start runs for real time deployment configurations. No one, even with this permission, can do any of the following. • Delete and update audience levels, data sources, named value lists, server groups, or deployment configurations created by the web service. • Create and deploy deployment configurations created by the web service.

Related tasks:

"Creating a role" on page 25

"Modifying role permissions" on page 25

"Removing a role from the system" on page 26

"Assigning a role to or removing a role from a group" on page 26

"Assigning a role to or removing a role from a user" on page 27

Chapter 4. Configuration management

When IBM Marketing Software is first installed, the Configuration page shows only the properties used to configure IBM Marketing Platform and some global configuration properties. When you install additional IBM Marketing Software applications, the properties used to configure these applications are registered with Marketing Platform. These properties are then shown on the Configuration page, where you can set or modify their values.

Some applications might have additional configuration properties that are not stored in the central repository. See application documentation for complete information about all configuration options for the application.

Related concepts:

Chapter 21, "Configuration properties on the Configuration page," on page 239

Property categories

The **Reports**, **General**, and **IBM Marketing Platform** categories are present when Marketing Platform is first installed. These categories contain properties that apply across all IBM Marketing Software applications installed in a suite.

- The default locale setting
- The **Security** category and sub categories with properties that specify login modes and mode-specific settings.
- Password settings
- Properties that configure data filters
- Properties that configure schedules
- Properties that configure the reporting feature
- Properties that configure how alerts are handled

Depending on the IBM Marketing Software applications that are installed, additional categories contain application-specific categories and sub categories. For example, after Campaign is installed, the **Campaign** category contains Campaign-related properties and sub categories.

Category types

A category can be one of three types, which are identified by different icons.

Table 13. Icons for category types

Category type	Icon
Categories that contain no configurable properties	
Categories that contain configurable properties	
Template categories that you can use to create a category Names of template categories are also shown in italics and enclosed in parentheses.	

© Copyright IBM Corp. 1999, 2018

Templates for duplicating categories

The properties for an IBM Marketing Software application are registered with Marketing Platform when the application is installed. When an application requires users to create duplicate categories for configuration purposes, a category template is provided.

To create a category, you duplicate the template. For example, you can create a new Campaign partition or data source by duplicating the appropriate template.

You can also delete any category that was created from a template.

Related concepts:

"Category naming restrictions"

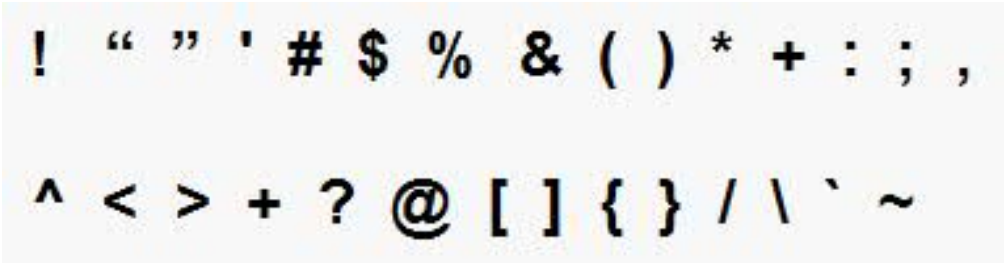
Related tasks:

"Creating a category from a template" on page 36

Category naming restrictions

The following restrictions apply when you name a category that you create from a template.

- The name must be unique among categories that are siblings in the tree (that is, among categories that share the same parent category).
- The following characters are not allowed in category names.



Also, the name cannot start with a period.

Related concepts:

"Templates for duplicating categories"

Related tasks:

"Creating a category from a template" on page 36

Property descriptions

You can access property descriptions in either of the following ways.

- Click **Help > Help for this page** to launch online help and display a topic that describes all of the properties for the page you are viewing.

- Click **Help > Product documentation** to launch a page that gives you access to all of the product documentation in online or PDF format. All property descriptions are included as an appendix in the *IBM Marketing Platform Administrator's Guide*.

The refresh function

A refresh button located at the top of the Configuration navigation tree provides the following functions.

- Refreshes the contents of the tree, which is useful you want to obtain the latest information about configuration settings. These settings might have been updated while you are viewing the tree (for example, when an application has been registered or unregistered or when someone else has updated settings).
- Returns the navigation tree to the state it was in the last time you selected a node, collapsing or expanding the tree as necessary.

  Important: If you are in edit mode when you click **Refresh**, the page is returned to the read mode. Any unsaved changes are lost.

The default user locale preference

Marketing Platform contains a default locale attribute that applies to all IBM Marketing Software applications that implement it.

You can set this default by setting the value of the **Region setting** property in the **Platform** category.

For details on this property, see its online help in the Configuration area or the *Marketing Platform Administrator's Guide*. To learn whether an IBM Marketing Software application implements this attribute, see the documentation for that application.

In addition, you can override these default values on a per-user basis by changing the value of this property in the user's account.

Related concepts:

"Locale preference" on page 13

Navigating to a category

Use this procedure to navigate to a category on the Configuration page.

1. Log in to IBM Marketing Software.
2. Click **Settings > Configuration** in the toolbar.

   The Configuration page shows the Configuration Categories tree.
3. Click the plus sign beside a category.

   The category opens, showing sub categories. If the category contains properties, they are listed along with their current values.

   The internal names for the categories are displayed under the page title. You use these internal names when you manually import or export categories and their properties using the `configTool` utility.

4. Continue to expand the categories and sub categories until the property you want to edit appears.

Editing property values

Use this procedure to modify a property value on the Configuration page.

1. Navigate to the category that contains the property you want to set.

   The Settings page for the category shows a list of all the properties in the category and their current values.
2. Click **Edit settings**.

   The Edit settings page for the category shows the property values in editable fields.
3. Enter or edit values as needed.

   In UNIX, all file and directory names are case-sensitive. The case of any file and folder name you enter must match the case of the file or folder name on the UNIX machine.
4. Click **Save changes** to save your changes or **Cancel** to exit the page without saving.

Related tasks:

"Navigating to a category" on page 35

Creating a category from a template

Use this procedure to create a category from a template on the Configuration page.

1. On the Configuration page, navigate to the template category you want to duplicate.

   Unlike other categories, template category labels are in italics and enclosed in parentheses.
2. Click the template category.
3. Enter a name in the **New category name** field (required).
4. You can edit properties within the new category now, or later.
5. Click **Save changes** to save the new configuration.

The new category appears in the navigation tree.

Related concepts:

"Templates for duplicating categories" on page 34

"Category naming restrictions" on page 34

Deleting a category

Use this procedure to delete a category on the Configuration page.

On the Configuration page, some categories can be deleted and others cannot. Any category you create from a template can be deleted. In addition, when an IBM Marketing Software product is registered, its set of categories might include categories that can be deleted.

1. On the Configuration page, navigate to the category you want to delete and click to select it to open its Settings page.

   If the category you have selected can be deleted, you see a **Delete Category** link.
2. Click the **Delete category** link.

   A window shows the message, Are you sure you want to delete "*category name*"?
3. Click **OK**.

The category no longer appears in the navigation tree.

Chapter 5. Dashboard management

Dashboards are configurable pages that contain information useful to groups of users who fill various roles within your company. The components that make up dashboards are called portlets. Dashboards can contain pre-defined portlets or portlets that you create.

You can create and configure dashboards yourself, or you can use the pre-assembled dashboards. Pre-assembled dashboards contain pre-defined portlets in combinations that are designed to be useful to users in a variety of roles within your organization.

You can also create your own custom portlets from IBM Marketing Software product pages, pages on your company intranet, or pages on the internet.

Related concepts:

"Custom portlets" on page 54

Dashboard planning

To plan how your organization uses the dashboard feature, you should work with your marketing management team to decide the following details.

- Which dashboards your users need.
- Which users should have access to which dashboards.
- Which portlets should go into each dashboard.
- Who should be designated as the dashboard administrator for each dashboard after the dashboards are rolled out. The dashboard administrator manages user access to the dashboard and modifies individual dashboard content and layout if necessary.

Dashboard audiences

You can control who views your dashboards by associating them with groups or by assigning individual users to them. Members of a group can access the dashboard or dashboards associated with that group, while non-members cannot view these dashboards.

You can also create one or more global dashboards, which can be seen by all IBM Marketing Software users within a partition regardless of their group membership or individual assignments.

When you create a global dashboard, you should include portlets that are of interest to the widest possible range of users. For example, if you have installed Campaign, you may want to include the My Custom Bookmarks portlet, one of the pre-defined IBM Marketing Software portlets.

© Copyright IBM Corp. 1999, 2018

User permissions required to view dashboards

Dashboards allow IBM Marketing Software users to view pages from multiple products (such as Marketing Operations and Campaign) in a single page, regardless of the permissions that are configured for them within those products.

Some dashboard portlets allow users to perform work in an IBM Marketing Software product by clicking a link within a portlet to open a page on which they can work. If the user does not have permissions to perform the task, the page does not display.

Some content within portlets is filtered based on the user. For example, if a user never works directly with campaigns, the My Recent Campaigns portlet might not display any links.

Pre-defined portlets

IBM Marketing Software provides two types of pre-defined dashboard portlets, which you can enable and then add to any dashboard you create.

IBM Marketing Software pre-defined portlets use Marketing Platform single-sign-on mechanism to access IBM Marketing Software content. Users are not prompted for credentials when they view a dashboard containing these portlets.

- List: A list of IBM Marketing Software items specific to the user. Examples of list portlets are My Recent Campaigns (Campaign), My Alerts (Marketing Operations, and the Continent Summary report (Digital Analytics for On Premises).
- IBM Cognos report: A specially formatted version of an IBM Marketing Software report.

You can also create your own custom dashboard portlets.

Related concepts:

"Custom portlet types and availability" on page 54

Pre-defined portlet availability

IBM Marketing Software provides pre-defined portlets with many of its products. Availability of the pre-defined portlets depends on the IBM Marketing Software products you have installed. Also, the IBM Cognos portlets are available only when the IBM Marketing Software reporting feature is implemented.

You must enable the pre-defined portlets in Marketing Platform before you can use them in a dashboard. IBM Marketing Software portlets are listed in Marketing Platform whether or not the product they belong to is installed. It is a good practice to enable only those portlets that belong to products that are installed. Only the portlets that are enabled appear in the list of portlets you can add to a dashboard.

Marketing Operations IBM Cognos report portlets

The following table describes the Marketing Operations dashboard portlets that are available after the Marketing Operations reports package is installed.

Table 14. Standard Marketing Operations IBM Cognos report portlets

Report	Description
Budget by Project Type	An example IBM Cognos report shows a 3-D pie chart of the budget per project type for the current calendar year. This report requires the Financial Management module.
Completed Projects by Quarter	An example IBM Cognos report shows a 3-D bar chart of the number of early, on-time, and late projects completed this quarter.
Forecast by Project Type	An example IBM Cognos report shows a 3-D pie chart of the forecasted spending per project type for the current calendar year.
Manager Approval Summary	An example IBM Cognos report shows data for active and completed approvals for all In Progress projects in the system.
Manager Task Summary	An example IBM Cognos report shows data for active and completed tasks for all In Progress projects.
Marketing Financial Position	An example IBM Cognos report shows a timeline with Budget, Forecasted, Committed, and Actual amounts for all plans in all states in the current calendar year. This report requires the Financial Management module.
My Task Summary	An example IBM Cognos report shows data about all active and completed tasks for the user who is viewing the report in all In Progress projects.
My Approval Summary	An example IBM Cognos report shows data about active and completed approvals for the user who is viewing the report.
Projects by Project Type	An example IBM Cognos report shows a 3-D pie diagram that shows all In Progress projects in the system by template type.
Projects by Status	An example IBM Cognos report shows a 3-D bar chart that shows all projects in the system by status: draft, in progress, on hold, canceled, and finished.
Projects Requested and Completed	An example IBM Cognos report shows a timeline graph of the number of project requests and number of completed projects per month. This report counts project requests with the following states only: Submitted, Accepted, or Returned.
Spend by Project Type	An example IBM Cognos report shows a 3-D pie chart of the actual amount that is spent per project type in the current calendar year. This report requires the Financial Management module.

Marketing Operations list portlets

If the Marketing Operations reports package is not installed, you still have access to the Marketing Operations list portlets that are available on your dashboard.

Table 15. Standard Marketing Operations list portlets

Report	Description
Approvals Awaiting Action	List of approvals that wait for your action.

Table 15. Standard Marketing Operations list portlets (continued)

Report	Description
Manage My Tasks	Lists your Pending and Active tasks and Not Started and In Progress approvals. An option to change the status of each item appears. • For tasks, you can change the status to Finish or Skip. • For Not Started approvals, you can change the status to Submit or Cancel. • For In Progress approvals that you own, you can change the status to Stop, Finish, or Cancel. • For In Progress approvals that you are assigned to approve, you can change the status to Approve or Reject.
My Active Projects	Lists your active projects.
My Alerts	Lists your Marketing Operations alerts.
My Project Health	Lists the name, health status, percentage complete, and number of tasks that are assigned to you for each project that you own or that includes you as a reviewer or member. The percentage complete is calculated as: `(Number of Finished Tasks + Number of Skipped Tasks) ÷ Total Number of Workflow Tasks` • To recalculate project health status, click . The system recalculates the health status for display by this portlet only. It does not work elsewhere in Marketing Operations. **Note:** Project health calculations can be made only at 5-minute intervals. • If you own more than 100 projects, click **Show All** to open the list in a new dialog. • To export listed project data into a .CSV file, click **Export**. • You can view the summary information for a project on the Summary tab. To view more metrics for project health, click the percentage complete indicator. To view the My Tasks list, click the number in the Tasks column.
My Requests	Lists requests that you own.
My Tasks	Lists tasks that you own.
Projects Over Budget	Lists all projects that are over budget for the calendar year. **Note:** This report requires the Financial Management module.

IBM Cognos report portlets for Campaign

The IBM Cognos report portlets are provided with the Campaign reports package. Use the report portlets to analyze response rates and campaign effectiveness.

You can enable and then add pre-defined dashboard portlets to any dashboard that you create. To manage your dashboards and add portlets to them, click **Dashboard > Create Dashboard**.

Table 16. IBM Cognos report portlets for Campaign

Report	Description
Campaign Return on Investment Comparison	An IBM Cognos report that compares, at a high level, the ROI of campaigns created or updated by the user viewing the report.

Table 16. IBM Cognos report portlets for Campaign (continued)

Report	Description
Campaign Response Rate Comparison	An IBM Cognos report that compares the response rates of one or more campaigns created or updated by the user viewing the report.
Campaign Revenue Comparison by Offer	An IBM Cognos report that compares the revenue received to date per campaign containing offers created or updated by the user viewing the report.
Offer Responses for Last 7 Days	An IBM Cognos report that compares the number of responses that were received over the last 7 days based on each offer created or updated by the user viewing the report.
Offer Response Rate Comparison	An IBM Cognos report that compares the response rate by offer created or updated by the user viewing the report.
Offer Response Breakout	An IBM Cognos report that shows the active offers created or updated by the user viewing the report, broken out by status.

Campaign list portlets

The standard Campaign list portlets are available for use on dashboards even if the reports package for Campaign is not installed.

Table 17. Campaign list portlets

Report	Description
My Custom Bookmarks	A list of links to websites or files created by the user viewing the report.
My Recent Campaigns	A list of the most recent campaigns created by the user viewing the report.
My Recent Sessions	A list of the most recent sessions created by the user viewing the report.
Campaign Monitor Portlet	A list of the campaigns that have run or are currently running that were created by the user viewing the report.

eMessage IBM Cognos report portlets

The following dashboard portlets are available in the eMessage reports package.

Report	Description
Recent Email Bounce Responses	This dashboard report presents data for various types of email bounces as a bar chart. The chart presents current bounce responses for the five most recent mailings that were sent before the current day.
Recent Email Campaigns Sent	This dashboard report provides a summary view of your most recent mailing activity. It lists totals for message transmission, recipient responses, and email bounces for the five most recent mailings that were sent before the current day.

Interact IBM Cognos report portlet

Interaction Point Performance - Shows the number of offers accepted per interaction point over a seven day period.

This dashboard report is defined to point to the interactive channel with the ID of 1. To create additional versions of this report (to report on additional interactive

channels) or to change the ID of the interactive channel that this report points to, see "How to configure the Interaction Point Performance dashboard portlet."

How to configure the Interaction Point Performance dashboard portlet

Interact has one IBM Cognos dashboard report: Interaction Point Summary. Because dashboard reports do not prompt users for query parameters, the channel ID of the interactive channel in the Interaction Point Performance report is a static value. By default, the channel ID for this report is set to 1. If the channel ID is not correct for your implementation, you can customize the report and change the channel ID in the report's filter expression.

To customize any of the IBM Cognos reports, you need IBM Cognos report authoring skills. For detailed documentation about creating and editing IBM Cognos BI reports, see the IBM Cognos BI documentation, especially *IBM Cognos BI Report Studio Professional Authoring User Guide* for your version of Cognos.

For information about the queries and data items in the Interaction Point Performance report, see the reference documentation that is provided in the Interact reports package.

To display a chart for more than one interactive channel in the Dashboard, make a copy of the Interaction Point Performance Dashboard and modify the channel ID. Then, create a new dashboard portlet for the new report and add it to your dashboards.

Distributed Marketing list portlets

This section describes the standard Distributed Marketing portlets that are available for use on dashboards.

Table 18. Distributed Marketing list portlets

Report	Description
List Management	A list of active Lists for the user viewing the report.
Campaign Management	A list of active Corporate Campaigns and On-demand Campaigns for the user viewing the report.
Subscription Management	A list of subscriptions to Corporate Campaigns for the current user.
Calendar	The Calendar showing the schedule for active Corporate Campaigns and On-demand Campaigns.

Contact Optimization list portlets

The standard Contact Optimization portlets that are available for use on dashboards.

Table 19. Contact Optimization list portlets

Report	Description
My recent Contact Optimization sessions	A list of the last 10 Contact Optimization sessions run by the user viewing the report within the last 30 days.
My recently successful Contact Optimization run instances	A list of the last 10 Contact Optimization sessions run by the user viewing the report that completed successfully within the last 30 days.

Table 19. Contact Optimization list portlets (continued)

Report	Description
My recently failed Contact Optimization run instances	A list of the last 10 Contact Optimization sessions run by the user viewing the report that did not complete successfully within the last 30 days.

Pre-assembled dashboards

IBM Marketing Software provides pre-assembled dashboards that include portlets appropriate for various audiences.

Pre-assembled dashboard availability

Pre-assembled dashboards are available as soon as you install Marketing Platform. However, to fully implement these dashboards you must also install any products required to support the portlets they include, and the portlets must be enabled.

For a pre-assembled dashboard to be available, at least one of the products that support it must be installed. For example, if a pre-assembled dashboard includes portlets that come from Campaign and eMessage, the dashboard will be available if either of these products is installed. If neither product is installed, the dashboard is not shown in the user interface. If one of the products is missing, the portlets that depend on that product are listed with a message indicating that they are not available.

List of pre-assembled dashboards

The following table describes the pre-assembled dashboards: their purpose, the portlets that comprise them, and the required products.

Table 20. List of pre-assembled dashboards

Pre-assembled dashboard	Purpose	Portlets	Required products
Campaign Management	This dashboard shows the financial results from campaigns.	• Financial Summary by Offer • Campaign Performance Comparison	• Campaign • Campaign Report Pack
Project and Traffic Management	This dashboard provides status updates for projects.	• My Tasks • My Alerts • My Active Projects • My Task Summary • Projects Requested and Completed • Approvals Awaiting Action • My Approval Summary • Projects by Status	• Marketing Operations • Marketing Operations Report Pack
Project Member	This dashboard shows tasks that require action and allows users to close completed tasks.	• My Tasks • My Active Projects • My Alerts • My Requests	Marketing Operations

Table 20. List of pre-assembled dashboards (continued)

Pre-assembled dashboard	Purpose	Portlets	Required products
Project Requests and Approvals	This dashboard shows tasks that require action, and provides status updates on projects and a high level overview of the marketing financial position and where funds are being spent.	• Approvals Awaiting Action • My Alerts • Marketing Financial Position • Projects by Project Type • Budget by Project Type • Spend by Project Type • Completed Projects by Quarter	• Marketing Operations with the Financial Management Module • Marketing Operations Report Pack
Project Financials	This dashboard provides a high level overview of the marketing financial position and where funds are being spent.	• Approvals Awaiting Action • Marketing Financial Position • Alerts • Projects by Type • Completed Projects by Quarter	• Marketing Operations with the Financial Management Module • Marketing Operations Report Pack

IBM Cognos report performance considerations

Reports are desirable components to add to dashboards because they add a visual element that makes it easy to scan large amounts of data. However, because reports require additional processing resources, performance can become an issue when many users access dashboards that contain many reports on a regular basis.

While organizations use data in different ways tailored to their needs, this section provides some general guidelines that should help you improve performance for dashboards that contain IBM Cognos reports. All of these guidelines apply to IBM Cognos report portlets, which are the most resource-intensive.

Scheduling runs in IBM Cognos

IBM Cognos reports can be scheduled to run at regular intervals. When a report is scheduled, it does not run every time a user accesses a dashboard containing that report. The result is improved performance of dashboards containing the report.

Only IBM Marketing Software reports that do not contain a user ID parameter can be scheduled in Cognos. When a report has no ID parameter, all users see the same data; the data is not filtered based on the user. The following portlets cannot be scheduled.

- All of the Campaign pre-defined portlets
- The Marketing Operations My Task Summary and My Approval Summary pre-defined portlets

Scheduling reports is a task that you perform in IBM Cognos; consult the Cognos documentation to learn more about scheduling in general. For specific scheduling requirements for dashboard portlets, see “Scheduling a dashboard report” on page 47.

Data considerations

You should plan scheduled runs based on the data contained in the report. For example, you would run the Offer Responses for Last 7 Days dashboard report every night so that it contains information relevant to seven days preceding the current day. In contrast, you might choose to run the Marketing Financials Position dashboard report once a week, because it compares financial indicators on a quarterly basis.

User expectations

An additional scheduling consideration is how frequently the intended users of the report expect the data to be updated. You should consult users about this when planning schedules.

Guidelines

Here are some broad guidelines to help you plan scheduling for dashboard IBM Cognos reports.

- Reports that include roll-up information should generally be scheduled to run every night.
- Reports that contain many calculations should be placed on a schedule.

Scheduling a dashboard report

To schedule a dashboard report (either a pre-defined or user-created portlet), you must first create a view and schedule it, and then configure the portlet as described here.

Note: You can schedule only those reports that are not filtered by user.

1. In Cognos, copy the report and save it under a new name.
2. In Cognos, open the copied report and save it as a view with the same name as the original report. Save it in the `Unica Dashboard/Product` folder, where `Product` is the appropriate product folder
3. In Cognos, schedule the view.
4. In IBM Marketing Software, add the report to the dashboard, if you have not done so already.
5. Only if the report is one of the pre-defined portlets, do the following in IBM Marketing Software.
   - On the Dashboard Administration page, click the **Edit portlet** icon next to the portlet.
   - Select **Yes** next to **Has this report been scheduled?**
   - Click **Save**.

Related tasks:

"Adding a pre-defined portlet to a dashboard" on page 51

"Adding a custom portlet to a dashboard" on page 57

Dashboard setup

Topics in this section describe how to set up dashboards.

Permissions required to administer dashboards

Only users with the Administer Dashboards permission in a partition can administer all of the dashboards in that partition. By default, this permission is granted to users with the AdminRole role in Marketing Platform.

When Marketing Platform is first installed, a pre-defined user, asm_admin, has this role for the default partition, partition1. See your administrator for the appropriate dashboard administrator credentials.

A user with the AdminRole role in Marketing Platform can assign any IBM Marketing Software user to administer individual dashboards in the partition to which that user belongs. Dashboard administration is done in the dashboard administration area of Marketing Platform.

Dashboard layout

The first time you add a portlet to a new dashboard, a window opens prompting you to select and save a layout. You can change the layout later by selecting the tab for the dashboard and selecting a different layout.

The options are as follows.

- 3 columns, equal width
- 2 columns, equal width
- 2 columns, 2/3-1/3 width
- 1 column, entire width
- Custom

Dashboards and partitions

If you are administering dashboards in a multi-partition environment, read this section to understand how multiple partitions affect dashboards.

In a multi-partition environment, a user can view or administer only those dashboards associated with the partition to which the user belongs.

When a dashboard administrator creates a dashboard, the following partition-related rules apply.

- Any dashboard that is created is available only to members of the same partition as the user who created it.
- Only those pre-defined portlets that are enabled in the partition to which the administrator belongs are available for inclusion in the dashboard.
- Only groups and users assigned to the same partition as the administrator are available for assignment to the dashboard.

Related concepts:

"Partitions and security management" on page 18

Related tasks:

"Overview of working with dashboards in a multi-partition environment" on page 49

Overview of working with dashboards in a multi-partition environment

When you have multiple partitions configured, follow these guidelines when you set up dashboards.

1. Before working with dashboards, associate one or more groups with each partition, and assign the appropriate users to each group.

   Only the platform_admin user, or another user with the PlatformAdminRole permissions can perform this task.
2. For each partition, ensure that at least one user has the Administer Dashboards permission, and make a note of these user names.

   The Marketing Platform AdminRole role has this permission by default, but you might want to create a role with more restricted access for dashboard administrators. These dashboard administrators can administer all dashboards within their partition.
3. For each partition configured in your system, do the following.
   a. Use an account that is a member of the partition and that can administer all dashboards in a partition to sign in to IBM Marketing Software.

      Refer to the list of users you created in the previous step.
   b. On the **Settings > Dashboard portlets** page, enable pre-defined portlets as needed.
   c. On the Dashboard Administration page, create the needed dashboards and add portlets.
   d. For each non-global dashboard, assign users who can view the dashboard.

      You can assign individual users or groups to the dashboard.
   e. For each dashboard, assign one or more users as dashboard administrator.

Related concepts:

"Dashboards and partitions" on page 48

Related tasks:

"Adding a pre-defined portlet to a dashboard" on page 51

"Enabling or disabling pre-defined portlets"

"Creating a dashboard that is not pre-assembled" on page 50

"Creating a pre-assembled dashboard" on page 50

Enabling or disabling pre-defined portlets

Perform this task before you begin to create dashboards. You should enable only those portlets that reference IBM Marketing Software products that you have installed.

1. Log in to IBM Marketing Software and select **Settings > Dashboard portlets**.
2. Click the check box next to portlet names to enable or disable them.

   A check mark enables a portlet, and clearing the check box disables a portlet.

   The portlets you selected are enabled and are available for inclusion in dashboards.

Related tasks:

"Overview of working with dashboards in a multi-partition environment"

Related reference:

"The Manage Portlets page" on page 53

Creating a dashboard that is not pre-assembled

Use this procedure to create a dashboard that is not pre-assembled

1. In IBM Marketing Software, select **Dashboard** to open the Dashboard administration page.

   All dashboards associated with your partition are shown.
2. Click **Create dashboard** to open the Create dashboard page.
3. Enter a unique title (required) and description (optional).
4. Select basic permissions.
   - If you want to restrict access to users who belong to a group associated with the dashboard, select **User or group-specific dashboard**.
   - If you want all users in the partition to be able to view the dashboard, select **Global dashboard for everyone**.
5. For the **Type** select **Create dashboard**.
6. Click **Save**.

   Your new dashboard appears as a tab on the Dashboard administration page, and is listed on the Administration tab.

You can now add portlets.

Related tasks:

"Overview of working with dashboards in a multi-partition environment" on page 49

Creating a pre-assembled dashboard

Use this procedure to create a pre-assembled dashboard.

1. Ensure that the portlets that comprise the pre-assembled dashboard you want to create are enabled.
2. In IBM Marketing Software, select **Dashboard** to open the Dashboard administration page.
3. Click **Create dashboard**.
4. For the **Type** select **Use pre-assembled dashboards**.

   The available pre-assembled dashboards are listed.
5. Select the pre-assembled dashboard you want to use and click **Next**.

   A list of the portlets comprising the selected pre-assembled dashboard is displayed. The list lets you know when a portlet is not available, either because the required product is not installed or because the portlet has not been enabled.
6. Click **Save** to finish creating the dashboard.

   Your new dashboard appears as a tab on the Dashboard administration page, and is listed on the Administration tab. You can now modify the portlets it contains, if necessary.

Related tasks:

"Enabling or disabling pre-defined portlets" on page 49

"Overview of working with dashboards in a multi-partition environment" on page 49

Adding a pre-defined portlet to a dashboard

Use this procedure to add a pre-defined portlet to a dashboard.

1. In IBM Marketing Software, select **Dashboard** and then select the tab for the dashboard you want to work with.
2. Click **Manage portlets** to view a list of enabled portlets.

   You can also access the Manage portlets page from the Administration tab, by clicking the Manage Portlets icon on the dashboard.
3. Select the check box next to one or more portlets to select it for addition to the dashboard.

   Use the following features to assist you in selecting portlets.
   - Filter the list of portlets by name or by the product that is the source of the portlet.
   - Display all portlets at once or page through the list.
   - Click column headings to sort the list alphabetically by source or portlet name, in ascending or descending order.
4. Click **Update**.

   The selected portlets are added to the dashboard.

Related concepts:

"Custom portlet types and availability" on page 54

Related tasks:

"Overview of working with dashboards in a multi-partition environment" on page 49

Related reference:

"The Manage Portlets page" on page 53

Removing a portlet from a dashboard

Use this procedure to remove a portlet from a dashboard.

1. In IBM Marketing Software, select **Dashboard**.

   A Dashboard Administration page opens. All dashboards associated with your partition are shown, with their portlets listed.
2. In the dashboard where you want to remove a portlet, click the **Delete** icon next to the portlet you want to remove.
3. Click **Yes, Delete** at the prompt.

   The portlet is removed from the dashboard.

Changing the name or properties of a portlet

Use this procedure to change the name or properties of a portlet.

1. In IBM Marketing Software, select **Dashboard**

   A Dashboard Administration page opens. All dashboards associated with your partition are shown, with their portlets listed.
2. In the dashboard you want to work with, click the **Edit Portlet** icon next to the portlet whose name you want to change.

   An Edit Portlet window opens.
3. Edit the name, description, URL, or hidden variables of the portlet.

4. Click **Save**.

Changing the name or properties of a dashboard

Use this procedure to change the name or properties of a dashboard.

1. In IBM Marketing Software, select **Dashboard**

   A Dashboard Administration page opens. All dashboards associated with your partition are shown.
2. In the dashboard you want to work with, click the **Manage Settings** icon at the bottom of the dashboard.

   A Settings tab opens.
3. Click the **Edit Dashboard** icon.

   An Edit Dashboard window opens.
4. Edit the title, description, or type of the dashboard, enable or disable it, or change whether users can change the layout..
5. Click **Save**.

Deleting a dashboard

Use this procedure to delete a dashboard.

1. In IBM Marketing Software, select **Dashboard**

   A Dashboard Administration page opens. All dashboards associated with your partition are shown.
2. In the dashboard you want to work with, click the **Delete Dashboard** icon at the bottom of the dashboard.
3. When prompted, click **Yes, Delete**.

   The dashboard is deleted.

Assigning or changing a dashboard administrator

Use this procedure to assign or change a dashboard administrator.

1. In IBM Marketing Software, select **Dashboard**

   A Dashboard Administration page opens. All dashboards associated with your partition are shown, with their portlets listed.
2. Click the **Manage Permissions** icon at the bottom of the dashboard you want to work with. A Manage Permissions tab opens.
3. Click the **Manage Dashboard Administrators** icon. A Manage Dashboard Administrators page opens. All dashboards associated with your partition are shown, with their portlets listed.
4. Select or deselect names.

   Users whose names are selected have administration permissions for the dashboard.

   You can do the following to find users.

   - Filter the list by entering all or part of a user name in the **Search** field.
   - Display all users, or only unassigned users, or only assigned users.
   - Sort the list by clicking column headings.
   - Display all users at once (based on your filtering criteria) or page through the list.
5. Click **Update**.

Related concepts:

"The dashboard administrator" on page 60

The Manage Portlets page

Refer to this table if you need help completing the fields in the Manage Portlets page.

Table 21. Fields on the Manage Portlets page

Field	Description
Filter	Enter part or all of a product name or portlet name to filter the portlet list based on the product that supplies the report or the name of the portlet.
Create Custom Portlet	Click to open a page where you can create a portlet that uses a URL you have obtained.
Create Quick Link Portlet	Click to open a page where you can create a quick link portlet.

Related tasks:

"Enabling or disabling pre-defined portlets" on page 49

"Adding a pre-defined portlet to a dashboard" on page 51

Quick link portlets

Quick links are pre-defined links to IBM Marketing Software products. Some quick links enable users to perform basic actions in the IBM Marketing Software product within the dashboard, without navigating to the product. You can configure portlets that contain a set of quick links that you choose.

Quick links for IBM Marketing Software products are installed when the product is installed. As of the 9.0.0 release, only Marketing Operations provides quick links. The same security considerations apply for quick links as for pre-defined portlets.

The following table describes the quick links available when Marketing Operations is installed.

Table 22. List of quick link portlets

Quicklink	Function
Create New Project Request	Opens up a popup window where you can choose a project template to create a Project Request. You can also click **Continue** to open the Project Request wizard in the application.
Create New Project	Opens up a popup window where you can choose a Project template to create a Project. You can also click **Continue** to open the Project wizard in the application.
Add Invoice	Opens the Add Invoice wizard in the application.
Projects	Opens the Project List page in the application.
Reports	Opens the **Analytics > Operational Analytics** page.
Resource Library	Opens the Asset Library page in the application.
Approvals	Opens the Approvals List page in the application.

Related tasks:

"Creating a quick link portlet" on page 54

Creating a quick link portlet

Use this procedure create a quick link portlet.

1. In the dashboard to which you want to add a quick link portlet, click **Manage Portlets**.

   A Manage Portlet page opens, listing the pre-defined portlets.
2. Click **Create Quick Link Portlet**.
3. Enter a portlet name and description, and select the quick links you want to include in the portlet.
4. Click **Save** to finish creating the portlet and to add it to the dashboard.

Related concepts:

"Quick link portlets" on page 53

Custom portlets

Topics in this section describe how to create and use custom portlets.

Custom portlet types and availability

You can create portlets from the following types of IBM Marketing Software pages.

- Any IBM Marketing Software IBM Cognos report, including Interact Interaction Point Performance reports that you have customized to point to additional interactive channels. You can customize any existing dashboard reports as described in this guide, or you can customize a non-dashboard report. For details on how to customize a non-dashboard report, see the *IBM Marketing Software Reports Installation and Configuration Guide*.
- Quick links portlets, which you can build using pre-defined links to IBM Marketing Software products.
- Any Digital Analytics for On Premises or Digital Analytics for On Premises On Demand report or dashboard that auto-updates.
- Any IBM Digital Analytics report.

In addition, you can create a portlet from a page on the internet or your company intranet.

Portlets that you create yourself are available for use in any dashboard. Your custom portlets are listed in the Manage Portlets window, where you can choose to add them to a dashboard.

Authentication considerations for custom portlets

When you are planning to create portlets, you should keep in mind the following authentication considerations.

- If your portlet is a Digital Analytics for On Premises report from an installation configured to use Marketing Platform for authentication or to use no authentication, or a dashboard report from any other IBM Marketing Software product that uses Marketing Platform for authentication, users are not prompted for credentials when they view the portlet.

- If your portlet is a Digital Analytics for On Premises report from an installation that is not configured to use Marketing Platform for authentication, the user must enter login credentials one time per browser session.
- If your portlet is a NetInsight OnDemand report or an internet or intranet page that requires authentication, the portlet behaves as a browser would. The user must enter login credentials in the content of the page the first time they view it during a browser session, and cookies are used to keep the user logged in.
- If your portlet is an IBM Digital Analytics report, users can view only those reports for which they have permissions in Digital Analytics. Also, if single-sign-on is enabled with Digital Analytics, users can view Digital Analytics reports in Marketing Platform dashboards without entering their credentials. Otherwise, users must enter their Digital Analytics credentials to view Digital Analytics reports in Marketing Platform dashboards.

Overview of the portlet creation process

This section provides an overview of the steps for creating a portlet, which are described in detail elsewhere in this guide.

See the related references if you need more information about performing this procedure.

1. Obtain and prepare the URL of the page you want to use as a portlet.

   To do this, you obtain the URL and modify it as needed.

   You can create portlets from the following sources.

   - Digital Analytics for On Premises report
   - IBM Cognos report
   - Digital Analytics report
   - NetInsight OnDemand report and pages on the internet or your company intranet
2. Add the URL to the `Platform_Admin_URL.properties` file.

   The `Platform_Admin_URL.properties` file is located in the `conf` directory under your Marketing Platform installation.
3. Stop and restart the Marketing Platform web application.
4. Add the portlet to a dashboard.

Related tasks:

"Preparing the URL from a Digital Analytics for On Premises report"

"Preparing the URL from an IBM Cognos dashboard report" on page 56

"Preparing the URL from a Digital Analytics report" on page 57

"Adding a custom portlet to a dashboard" on page 57

Preparing the URL from a Digital Analytics for On Premises report

Use this procedure for reports in a Digital Analytics for On Premises installation.

1. In Digital Analytics for On Premises, display the report you want to export.

   If you are using a Digital Analytics for On Premises dashboard, only the top left report on the dashboard is exported.

2. Click the **Export** icon located in the toolbar at the upper right of the report.

   The Export options window opens.
3. Complete the fields as follows.
   - Select **Portlet URL** from the **Export Type** drop-down.
   - Select `Web Browser` from the **Format of Report** drop-down.
   - Specify the number of values to include in the report.
   - Specify the width of the report graphic, in pixels. Path reports self-adjust their size, regardless of the width you specify. Stacked bar reports automatically increase the width you specify by 30%.
   - Choose to hide the report header, as the portlet has a title that you can edit.
4. Click **Export**.

   The report URL is displayed in a dialog box.
5. Copy the URL and paste it into a text editor.
6. Add the following to the beginning of the report URL:

   Your_IBM_Marketing_Software_URL`/suiteSignOn?target=`

   where *Your_IBM_Marketing_Software_URL* is the login URL for your installation of IBM Marketing Software.

   For example, suppose you have the following information.
   - Your report URL is `MyReportURL`
   - The login URL for your installation of IBM Marketing Software is `http://myHost.myDomain:7001/unica`

   Your final URL would be `http://myHost.myDomain:7001/unica/suiteSignOn?target=MyReportURL`

Related tasks:

"Overview of the portlet creation process" on page 55

Preparing the URL from an IBM Cognos dashboard report

The format of an IBM Cognos dashboard portlet URL is as follows.

For information about creating dashboard reports with IBM Cognos, see the *IBM Marketing Software Reports Installation and Configuration Guide*.

`http(s)://`*`HOST.DOMAIN`*`:`*`port`*`/unica/reports/jsp/dashboard_portlet.jsp?product=`*`Product`*`& report=`*`ReportName`*

where

- *`Product`* is the name of the IBM Marketing Software application's subfolder in the **Unica Dashboards** folder on the IBM Cognos system. That is: `Campaign`, `Interact`, or `Plan` for Marketing Operations. (Plan was the previous name of the Marketing Operations application.)
- *`ReportName`* is the name of the dashboard report. For example: Campaign Performance Comparison

For example,

```
http://serverX.example.com:7001/unica/reports/jsp/
dashboard_portlet.jsp?product=Campaign&report=Campaign Performance
Comparison
```

If you have scheduled the report, add the following to the end of the URL:

```
&isView=true
```

Related tasks:

"Overview of the portlet creation process" on page 55

"Scheduling a dashboard report" on page 47

Preparing the URL from a Digital Analytics report

Use this procedure for Digital Analytics reports.

If you want users to be able to view Digital Analytics reports in dashboards without having to log in to Digital Analytics, you must enable single sign-on between IBM Marketing Software and Digital Analytics.

1. Log in to Digital Analytics and navigate to the report that you want to add as a portlet.
2. Copy the URL shown in your browser.

   The link is copied to your clipboard and is ready to be pasted into the IBM Digital Analytics URL field in the Create Custom Portlet window in Marketing Platform.

   To ensure the URL is not overwritten should you copy something else before using it to create a portlet, you can paste it into a text editor.

Related concepts:

Chapter 15, "Implementation of one-way SSL," on page 153

Related tasks:

"Overview of the portlet creation process" on page 55

Related reference:

Chapter 10, "Single sign-on between IBM Marketing Software and IBM Digital Analytics," on page 109

Preparing the URL from an intranet or internet page

For portlets created from intranet or internet pages, including Digital Analytics for On Premises pages, point your browser to the desired page and copy the URL from your browser's address field.

Use the copied URL when you create your custom portlet.

Adding a custom portlet to a dashboard

Perform this procedure to add a custom portlet to a dashboard.

Before performing this procedure, you should have done the following.

- Prepared a URL as described elsewhere in this section
- Added the URL to the `Platform_Admin_URL.properties` file, which is located in the `conf` directory under your Marketing Platform installation

- Stopped and restarted the Marketing Platform web application

1. In IBM Marketing Software, select **Dashboard** and then select the tab for the dashboard you want to work with.
2. Click **Manage Portlets**.

   A **Manage Portlets** window opens.
3. Click **Create Custom Portlet**.

   A **Create Custom Portlet** window opens.
4. Do one of the following sets of steps, depending on the type of portlet you are adding.

   If you are creating a portlet that is not a Digital Analytics report portlet, do the following.
   - For the **Type**, select **Custom**.
   - Complete the **Name** and **Description** fields.
   - Paste the contents of your clipboard (which contains the URL you obtained earlier) into the **URL** field.

   If you are creating a Digital Analytics report portlet, do the following.
   - For the **Type**, select **IBM Digital Analytics**.
   - Complete the **Name** and **Description** fields.
   - Paste the contents of your clipboard (which contains the URL you obtained earlier) into the **IBM Digital Analytics URL** field.
5. Click **Save**.

   The window closes and you return to the Administration tab. The new portlet is located in the upper left corner, where it may overlay a previously added portlet. Click and drag the portlet heading to place the portlet in an appropriate position in the dashboard.

Related concepts:

"Dynamic tokens"

Related tasks:

"Overview of the portlet creation process" on page 55

Related reference:

"The Create Custom Portlet page" on page 59

Dynamic tokens

When you define a custom dashboard portlet, you can use pre-defined tokens that are replaced with the values stored in Marketing Platform for the current user when the portlet is invoked.

This feature is not available for custom portlets from Digital Analytics.

The following tokens are supported.

- `<user_name>`
- `<user_first_name>`
- `<user_last_name>`
- `<user_email>`

The URL is invoked with hidden variables passed as request parameters.

The values must be present in the user details in Marketing Platform. Also, you must know the names of the variables used by the target web site.

To use these tokens, enter the name value pairs in the **Hidden Variables** field of the Create Custom Portlet page. If you use multiple tokens, separate them with a semicolon.

For example, suppose you want to send a user's first and last name in a portlet URL. In this example, the receiving web site expects `fname` and `lname` to contain the user's first and last names respectively. You would complete the **URL** and **Hidden Variables** fields as follows.

- **URL** - `www.example.com`
- **Hidden Variables** - `fname=<user_first_name>;lname=<user_last_name>`

Related tasks:

"Adding a custom portlet to a dashboard" on page 57

Related reference:

"The Create Custom Portlet page"

The Create Custom Portlet page

Refer to this table if you need help completing the fields on the Custom Portlet page.

Table 23. Fields on the Create Custom Portlet page

Field	Description
Type	Select the portlet type: a portlet that is not from Digital Analytics, or a portlet that is from Digital Analytics.
Name	Enter an appropriate name for the portlet.
Description	Enter a description for the portlet that lets other administrators know why it is part of this dashboard.
URL or Digital Analytics URL	Paste in your prepared URL.
Hidden Variables	Available only when the portlet is not from Digital Analytics. If your portlet requires users to log in, you can enter name/value pairs to securely send these credentials to the site. You must obtain the expected variable name from the web site.

Related concepts:

"Dynamic tokens" on page 58

Related tasks:

"Adding a custom portlet to a dashboard" on page 57

Dashboard membership administration

Topics in this section describe how to manage dashboard membership.

The dashboard administrator

If you have been designated a dashboard administrator, you are responsible for managing the membership, layout, and content of that dashboard. This section describes how to manage dashboard membership.

Related tasks:

"Assigning or changing a dashboard administrator" on page 52

"Granting or removing dashboard membership"

Granting or removing dashboard membership

Use this procedure to grant or remove dashboard membership.

1. In IBM Marketing Software, select **Dashboard** and then select the tab for the dashboard you want to work with.
2. Click the **Manage Permissions** icon at the bottom of the dashboard you want to work with.

   A Manage Permissions tab opens.
3. Click the **Manage Dashboard Users** icon.

   A Manage Dashboard Users page opens.
4. Select or deselect the checkbox to grant or remove access to the dashboard.

   Users whose names are selected can view the dashboard.

   You can do the following to find users.
   - Filter the list by entering all or part of a user name in the **Search** field.
   - Display all users, or only unassigned users, or only assigned users.
   - Sort the list by clicking column headings.
   - Display all users at once (based on your filtering criteria) or page through the list.
5. Click **Update**.

Related concepts:

"The dashboard administrator"

Chapter 6. The IBM Marketing Software Scheduler

The IBM Marketing Software Scheduler enables you to configure a process to run at intervals that you define.

Items that you can schedule

You can schedule the following.

- Campaign flowchart runs

  Note: The IBM Marketing Software Scheduler is completely independent of the Schedule process in Campaign.
- Contact Optimization optimization session and post-optimization flowchart runs
- eMessage mailings
- Marketing Operations bulk deactivations
- Calls to external APIs
- IBM Marketing Software alerts and notifications
- External batch or shell scripts

Schedules and runs

The scheduler uses two basic concepts: schedules and runs.

- A schedule is any task that you want to run once or on a recurring basis. When you define a schedule you specify the IBM Marketing Software object, the start and end dates, and optionally, the frequency with which the task is run (called a recurrence pattern).
- A run is an execution instance of a schedule.

Types of schedules

There are three types of schedules.

- Time-based - Runs occur at specified times.
- Trigger-based - Runs occur when a schedule receives a specified trigger (for example, when another schedule sends a trigger on success or failure of its run, or when the scheduler utility sends a trigger).
- Multiple-run-based - Runs are dependent on other schedules, and occur only when multiple other schedules have finished their runs

Schedule notifications

You can set up notifications that are sent to yourself for schedules you create, and administrators can set up notifications that are sent to groups of users for schedules created by anyone.

Related concepts:

"Difference between the IBM Campaign Schedule process and IBM Marketing Software Scheduler" on page 77

© Copyright IBM Corp. 1999, 2018

Scheduler triggers that are sent on success or failure of runs

When you create or edit a schedule, you can configure a trigger that the schedule sends on success or failure of a run, and you can also configure one or more schedules to listen for these triggers.

Triggers work across products. For example, a Campaign flowchart can send a trigger that starts an eMessage mailing.

A trigger is a text string that the IBM Marketing Software Scheduler can send when a run completes successfully or when a run fails. Each schedule can send one trigger on successful conclusion of a run, and one trigger on failure of a run. Also, each schedule can listen for one success and one failure trigger.

All schedules set to listen for a trigger receive all sent triggers, but a schedule initiates a run only if it receives the trigger for which it is listening. An unlimited number of dependencies between schedules can be created in this manner.

After you have created a trigger, it appears in a dropdown list of triggers in the scheduler user interface, which makes it easy to use again.

Trigger example

You can schedule a set of Campaign flowcharts to run at the same time by configuring them to all listen for the same trigger, which can be sent by any other schedule or by an external application using the “scheduler_console_client” on page 233 utility. You can also use triggers to cause a set of flowcharts to run in series, one after another.

The following example illustrates how to set up a series of flowcharts to run in a specified order.

- Flowchart 1 is scheduled with a "Flowchart 1 run complete" trigger that is sent when the run completes successfully.
- Flowchart 2 is scheduled as follows.
  - Start when a "Flowchart 1 run complete" trigger is received.
  - Send a "Flowchart 2 complete" trigger when the run completes successfully.
- Flowchart 3 is scheduled to start when a "Flowchart 2 run complete" trigger is received.

About start triggers

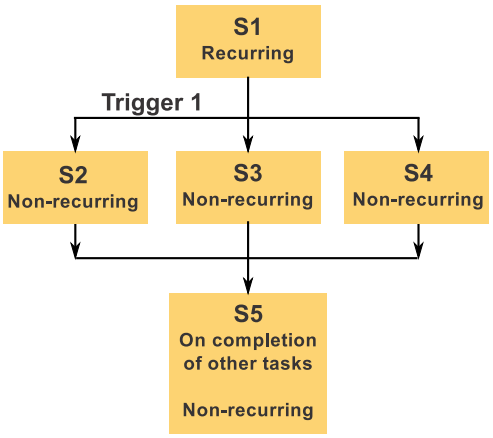
A schedule that is set up with a start trigger begins to listen for a trigger as soon as it is created, regardless of its start date. However, the trigger does not override the start date. For example, if a schedule has a start date of December 12, 2016 and on December 5, 2016 it receives its start trigger, the run does not start until December 12, 2016.

Schedules that depend on completion of multiple runs

You can configure a schedule to run only when multiple other schedules have finished their runs by using the **On completion of other tasks** option in the **When to start** drop down list.

For example, suppose you have a schedule, S1, that is set up with a recurrence pattern. S1 has a trigger that is sent every time an S1 run completes successfully.

Three schedules, S2, S3, and S4, are configured to start when they receive the outbound trigger from S1. You can set up an additional schedule, S5, that runs when S2, S3, and S4 complete successfully. S5 runs only when all three of the runs on which it is dependent complete. The following diagram illustrates this example.



To set up a scenario like the one described in the example, you would configure S5 using the **On completion of other tasks** option in the **When to start** drop down list.

When you configure a run to be dependent on other runs in this way, you must keep in mind the following considerations.

- The schedules on which the schedule you are configuring depends must be non-recurring. In the example above, S2, S3, and S4 must be non-recurring. However, because S1 recurs, S2, S3, and S4 effectively recur, based on S1 runs.
- The schedule that is dependent on other schedules must also be non-recurring. In the example, S5 must be non-recurring. Again, because S1 recurs, S5 effectively recurs as well.
- The schedule that is dependent on other schedules cannot be used as one of the criteria in the **On completion of other tasks** option for any other schedule. In the example, S5 cannot be used as a criterion in the **On completion of other tasks** option for any other schedule.
- If you want to delete a schedule that is configured with the **On completion of other tasks** option, you must first change the configuration to remove the **On completion of other tasks** option. Then you can delete the schedule.

Schedule triggers that are sent from an external script

The IBM Marketing Software Scheduler can respond to triggers sent by an external application. The `scheduler_console_client` utility enables this feature. This utility issues triggers that can launch one or more schedules set up to listen for that trigger.

Because `scheduler_console_client` is a batch script application, it can be called by external applications, possibly using another batch script.

For example, if you set up a schedule that is listening for a trigger "T1," you could run the `scheduler_console_client` utility with the following command to send the T1 trigger: `scheduler_console_client.bat -v -t T1`

The utility can provide the following information.

- A list of the schedules that are configured to listen for any given trigger.
- Whether it has successfully sent the trigger. Note that the utility cannot report whether the schedule that is listening for the trigger executed successfully. That information is available on the scheduler management pages.

You can not use this utility to set up a schedule to listen for a trigger or to modify a trigger for which a schedule is listening. You must perform these actions in the scheduler user interface.

Example script

Here is an example of a script to that causes the `scheduler_console_client` utility to issue the string "example_trigger". This trigger would set off a run of a schedule set up to listen for "example_trigger".

You could call a script like this from an external application when that application generates an event.

The example script assumes that the script is in the same directory as the utility.

```
@rem*****************************************************
@rem This script is used to call the Marketing Platform
@rem scheduler_console_client.
@rem*****************************************************

echo Now starting scheduler trigger.
set JAVA_HOME=c:\jdk15_12
call scheduler_console_client.bat -v -t example_trigger

@rem*****************************************************
```

Security considerations

Scheduling within the enterprise applications is considered to be an administrator's activity. It is assumed that any user who has execute permission in the host operating system for the `scheduler_console_client` utility is also authorized to issue triggers.

To prevent any user from using this utility to issue a trigger, you should revoke execute permission for the `scheduler_console_client` utility for that user.

Related reference:

"scheduler_console_client" on page 233

Scheduler recurrence patterns

You can set up a schedule to run repeatedly by configuring a recurrence pattern. Any recurrence pattern you set begins after the start time you specify.

You have several recurrence pattern options.

- Pre-defined - A set of common recurrence patterns from which you can select
- Cron expression - A string composed of 6 or 7 fields separated by white space that represents a set of times

- Simple custom recurrence pattern - A user interface for creating recurring patterns that is similar to many common meeting schedulers

All of the scheduler recurrence patterns are based on cron expressions. The scheduler provides pre-defined patterns in the user interface for easier creation of these cron expressions. If you write your own custom cron expression, it is a good practice to provide a meaningful description of the recurrence pattern, to make it easier for anyone who is not fluent in reading these expressions to understand the pattern.

Important: All of the recurrence patterns reset at the end of the next longer interval. For example, if you set a custom weekly pattern to run every three weeks, it runs the third week of every month, because the pattern resets at the end of every month. This is a characteristic of all cron expressions. To set a schedule that actually runs on week 3, 6, 9, 12, and so on, you must create separate schedules for each desired execution date.

Time zone support

You can schedule runs to occur in the context of any one of a large number of worldwide time zones.

When you create a schedule, the default is always the time zone of the server on which Marketing Platform is installed. However, you can select from any other time zones listed in the **Select time zone** drop down list. These options are expressed as GMT times followed by the commonly used term for that time zone. For example, (GMT-08:00) Pitcairn Islands or (GMT-08:00) Pacific Time (US & Canada).

The selected time zone is applied to all aspects of the schedule, including the following.

- Information shown on the Schedules and Runs tabs
- Recurrence patterns and triggers

Scheduler throttling

Throttling is used to manage performance when a large number of processes are likely to place high demands on the system. Throttling is based on scheduler groups that you set up on the **Settings > Configuration** page. You assign a throttling threshold to a group, and associate schedules with that group.

The throttling threshold is the highest number of runs associated with that group that can run concurrently. To reduce resource consumption on the server, you can set the throttling threshold to a smaller value. Only schedules created in the IBM Marketing Software Scheduler are subject to throttling.

Unlimited threshold in the default group

All schedules must belong to a throttling group. If you do not want to enable throttling for a schedule, make it a member of the Default scheduler group (the default selected option in the **Scheduler Group** field when you create a schedule). This group has a high throttling threshold, which effectively means that no throttling is in place.

Throttling exception

If you run a flowchart from within Campaign or by using the Campaign `unica_svradm` utility, these runs do not count in the throttling threshold, and they begin execution immediately.

Throttling examples

- If system resources are a concern, you can use throttling to manage the load on a server. For example, if many complex Campaign flowcharts must be run, you can assign them to a throttling group that limits the number of flowcharts that can be run at the same time. This throttling helps to manage the load on the Campaign server or the marketing database.
- You can use throttling to set priorities for schedules. By assigning high-priority schedules to a group with a high throttling threshold, you ensure that runs of these schedules are performed using system resources as efficiently as possible. You should assign lower-priority schedules to groups with lower throttling thresholds.
- If you have a flowchart that is scheduled with a recurrence pattern, you can use throttling to ensure that runs occur in sequence, without overlapping. For example, suppose you have scheduled a flowchart with a recurrence pattern set to execute a run every hour for 10 hours. If the flowchart takes more than one hour to complete a run, the next run could attempt to begin before the previous run is completed, resulting in failure because the still running flowchart would be locked. To ensure that this does not happen, you can create a throttling group with a threshold of 1, and assign the flowchart's schedule to this group.

Related tasks:

"Setting up throttling for the IBM Marketing Software Scheduler"

Setting up throttling for the IBM Marketing Software Scheduler

You must set up a throttling group for each type of object being scheduled.

1. On the Configuration page, navigate to one of the throttling group templates under `Platform > Scheduler > Schedule registrations > [Product] > [Object] > Throttling group`.
2. Create a category from the throttling group template.

   The number you set for the `Throttling threshold` property is the highest number of runs associated with that group that can execute concurrently. Any schedules eligible to run that exceed the throttling threshold are queued to run in the order in which the run notification is received by the scheduler.

   The configured scheduler groups appear in the **Scheduler Group** drop-down list in the Scheduler user interface for creating and editing schedules.

   You must create a throttling group for each type of object whose runs you want to control in this way. For example, flowchart throttling groups are available only for scheduling flowcharts; mailing throttling groups are available only for scheduling mailings.
3. Assign one or more schedules to the group, as needed.

Related concepts:

"Scheduler throttling" on page 65

Related tasks:

"Creating a category from a template" on page 36

Whitelist prerequisite for external tasks (with FixPack 10.0.0.1 only)

Only if you have applied Marketing Platform FixPack 10.0.0.1, a whitelist prerequisite applies to any external tasks that you create to schedule API calls or scripts.

Before you can schedule an external task, you must add the API or script to a whitelist located in the `conf` directory under your Marketing Platform installation.

Related tasks:

"Adding an API to the whitelist" on page 68

"Adding a script to the whitelist"

Related reference:

"The create a schedule wizard" on page 69

Adding a script to the whitelist

Only if you have applied Marketing Platform FixPack 10.0.0.1, perform this procedure before you create any external tasks that schedule a script.

The script must be on the web application server where Marketing Platform is deployed.

1. Open the whitelist file for scripts in a text editor.

   The whitelist file for scripts is `Platform_Admin_Scheduler_Scripts.properties`. This file is located in the `conf` directory under your Marketing Platform installation.

2. Enter the full path of the batch or shell script you plan to schedule, and include the number of parameters that are used in the script you are scheduling.

   For example, suppose you want to schedule a script that is named `RunETLJobs.bat` and that takes these three parameters: *username*, *password*, *db_table*.

   You would make the following entry in the whitelist file. The entry includes the absolute path of the script, followed by a space and the number of parameters used. The parameter count must exactly match the number of parameters that are used in the scheduled script.

   ```
   C:\Scripts\RunETLJobs.bat 3
   ```

   When you create the schedule, in the **Run parameters** field you specify the parameter names between double number signs (##) followed by a space, as shown in the following example.

   ```
   C:\Scripts\RunETLJobs.bat ##username## ##password## ##db_table##
   ```

3. Save and close the whitelist file.

   Now you can schedule the script on the Schedules tab of the **Settings > Schedule management** page.

Related concepts:

"Whitelist prerequisite for external tasks (with FixPack 10.0.0.1 only)"

Related reference:

"The create a schedule wizard" on page 69

Adding an API to the whitelist

Only if you have applied Marketing Platform FixPack 10.0.0.1, perform this procedure before you create any external tasks that schedule an API call.

1. Open and edit the whitelist file for APIs in a text editor.

   The whitelist file for APIs `Platform_Admin_Scheduler_API.properties`. This file is located in the in the `conf` directory under your Marketing Platform installation.

2. Enter the URI of the API you plan to schedule, and if query parameters are used, include these parameter names, without including values.

   For example suppose you want to schedule the following API call, using all of the query parameters shown.

   `http://www.example.com/tickets?fields=id&state=open&sort=updated_at`

   You would make the following entry in the whitelist file, listing all of the parameters.

   `http://www.example.com/tickets?fields&state&sort`

   With this whitelist entry, you can schedule API calls that use some or all of the listed parameters. For example:

   - `http://www.example.com/tickets`
   - `http://www.example.com/tickets?fields=id`
   - `http://www.example.com/tickets?fields=id&state=open`
   - `http://www.example.com/tickets?fields=id&state=open&sort=updated_at`
   - `http://www.example.com/tickets?fields=id&sort=updated_at`
   - `http://www.example.com/tickets?fields=id&state=open`

   API calls that use non-listed query parameters cannot be scheduled. Scheduler validation fails if any parameters are used that are not present in the whitelist.

3. Save and close the whitelist file.

   Now you can schedule the API call on the Schedules tab of the **Settings > Schedule management** page.

Related concepts:

"Whitelist prerequisite for external tasks (with FixPack 10.0.0.1 only)" on page 67

Related reference:

"The create a schedule wizard" on page 69

Best practices for setting up schedules

These are some best practices for planning and configuring scheduled runs of IBM Marketing Software objects.

For optimal performance and ease of maintenance, keep these guidelines in mind.

- Because scheduled runs are executed on the system where the client product is installed, consider the scaling capabilities of the client system. Stagger runs or use throttling to tune the system.
- When possible, schedule heavy jobs during low system load times.
- Avoid overlapping runs, which cause run failures.
  - Use caution if you use the same object in multiple schedules. For example, if you use flowchart F1 in three schedules, these schedule definitions could cause a run to be started before a previous run completes, causing run failure.

- If a flowchart run is initiated manually or by an external script, a subsequent attempt to run the flowchart by any means fails with a lock error if the previous run has not completed.

- The scheduler creates large quantities of data. If you observe performance issues with the scheduler, consider removing schedule definitions that you no longer need.

  Important: Removing a schedule definition also removes its associated run history from the database.

The create a schedule wizard

This section describes in detail the pages you use when you create a schedule.

The following table describes the fields you use when you schedule runs of Campaign flowcharts, eMessage mailings, Contact Optimization sessions, external scripts, and API calls.

Table 24. Fields in the create a schedule wizard

Field	Description
Select a task type	The type of object to be scheduled. The following options are available. • **External task - script** Allows you to schedule invocation of tasks defined in batch or shell scripts external to IBM Marketing Software. Only if you have applied Marketing Platform FixPack 10.0.0.1, the script must be listed in a whitelist file located in the `conf` directory under your Marketing Platform installation. Also, the script must be on the web application server where Marketing Platform is deployed. • **External task - API** Allows you to schedule invocation of APIs external to IBM Marketing Software. Only if you have applied Marketing Platform FixPack 10.0.0.1, the API must be listed in a whitelist file located in the `conf` directory under your Marketing Platform installation. • **Campaign flowchart** Allows you to schedule invocation of Campaign flowcharts. Selecting this option takes you to the Campaign list page where you select a campaign, optionally set flowchart override parameters, and schedule the flowchart run. • **Contact Optimization session** Allows you to schedule invocation of Contact Optimization sessions. Selecting this option takes you to the Contact Optimization sessions list page where you select a session and schedule the session. • **eMessage mailing** Allows you to schedule invocation of eMessage mailings. Selecting this option takes you to the eMessage mailings list page where you select and schedule the mailing. • **Marketing Operations bulk deactivation** Allows you to schedule bulk deactivation of projects in Marketing Operations. Selecting this option takes you to the Marketing Operations Administrative Settings page where you click **Deactivation Administration** and schedule the bulk deactivation. • **Alert** Allows you to schedule alerts for users of IBM Marketing Software. Selecting this option opens a window where you set the message title, message body, and severity. After you click **Schedule this alert**, you can create a schedule for the alert. Users can manage their notification subscriptions based on severity. • **Notification** Allows you to schedule notifications for users of IBM Marketing Software. Selecting this option opens a window where you set the message title, message body, and severity. After you click **Schedule this notification**, you can create a schedule for the notification. Users can manage their notification subscriptions based on severity.
Schedule name	Enter a name for the schedule.
Scheduler group	If you have created one or more throttling groups, you can associate this schedule with a group to limit the number of runs of this schedule that can execute at the same time. Throttling groups configured on the Configuration page appear as options in this field
Description	Enter a description for the schedule.

Table 24. Fields in the create a schedule wizard (continued)

Field	Description

Run parameters	Used when you schedule APIs and scripts.
	
	Only if you have applied Marketing Platform FixPack 10.0.0.1, a whitelist prerequisite applies to any external tasks that you create to schedule API calls or scripts. Before you can schedule an external task, you must add the API or script to a whitelist located in the `conf` directory under your Marketing Platform installation.
	• For API schedules, enter the URI, plus any parameters in the format shown in the examples.
	API with no parameters: `http://example.com`
	API with parameters: `http://www.example.com/tickets?fields=id&state=open &sort=updated_at`
	Currently, there is no support for Marketing Platform tokens in the URI.
	• For script schedules, enter the full path to the script on the Marketing Platform server, plus any parameters in the format shown in the examples. Specify the parameter names between double number signs (##) followed by a space.
	– Windows examples
	Script with no parameters: `C:\Scripts\ExecuteDatabaseJob.bat`
	Script with parameters:
	`C:\Scripts\RunETLJobs.bat ##username## ##password## ##db_table##`
	– UNIX examples
	Script with no parameters: `/opt/ExecuteDatabaseJob.sh`
	Script with parameters:
	`/opt/RunETLJobs.sh ##username## ##password## ##db_table##`
	
	Execution of these tasks is asynchronous. Marketing Platform does not track success or failure of script and API tasks. Status indicates only whether they are launched successfully.

On successful completion, send a trigger	If you want runs of this schedule to send a trigger when the run completes successfully, enter the trigger text here. Other schedules can be set to listen for this trigger.
On error, send a trigger	If you want runs of this schedule to send a trigger when the run fails, enter the trigger text here. Other schedules can be set to listen for this trigger.
Search tags / keywords	Enter any tags you want you associate with the schedule for use in searches. Separate multiple entries with commas.
Schedule status	Whether the schedule is enabled or disabled. Disabling a schedule applies only to future or queued runs of that schedule. Any run currently underway is not affected. The default status is **Enabled**.
Select time zone	If you select an option other than the server default, the **Start**, **End**, and **Last updated** columns on the Schedule management page show both the server default time and the time in the selected zone.

Table 24. Fields in the create a schedule wizard (continued)

Field	Description
When to start	Select one of the following options to specify the first time the schedule runs. The start time applies only to the first run; it defines the time when a schedule is first eligible to run. The actual first run might be after the start date if any of the following conditions are present. • The schedule is configured to wait for a trigger. • The schedule is a member of a throttling group. • The schedule uses a recurrence pattern. • **Now** • **On a date and time** - Select a date and time. • **On a trigger** - Select an existing trigger or enter a new one. If you enter a new one, you must configure a schedule to send this same string on success or failure. • **On a trigger after a date** - Select an existing trigger or enter a new one, and select a date and time. If you enter a new trigger, you must configure a schedule to send this same string on success or failure. • **On completion of other tasks** - Select from a list of existing schedules. The schedule runs only when the selected other schedules have finished their runs.
Number of runs	Select one of the following options to specify the number of runs. • **Run once only** - The schedule runs one time. It is eligible to execute the run on the start date and time you specify. • **Stop after n occurrences** - Runs stop after the specified number of runs have occurred (whether the runs succeed or fail) or the end date arrives, whichever is first. • **Stop by a date and time** - Runs are initiated as many times as defined until the specified end date and time is reached. A run might execute after this time if the run execution has been delayed due to throttling constraints. • **On completion of other tasks** - The schedule runs only when all the other tasks selected for this option complete successfully. When you click the **Set up recurrences** button, you can select one of the following options. • **Use a pre-defined recurrence pattern** - Select a pattern from the list. The Marketing Platform provides a set of pre-defined patterns, and you can create your own by adding properties on the Configuration page. • **Use a simple custom recurrence pattern** - Select an interval. • **Use a cron recurrence expression** - Enter a valid cron expression.

Related concepts:

"Whitelist prerequisite for external tasks (with FixPack 10.0.0.1 only)" on page 67

"Time zone support" on page 65

"Schedules that depend on completion of multiple runs" on page 62

Related tasks:

"Adding an API to the whitelist" on page 68

"Adding a script to the whitelist" on page 67

Related reference:

"The Schedule management pages" on page 82

Run exclusions

10.0.0.2 From the 10.0 Fix Pack 2 release onwards, you can create exclusion rules to exclude the scheduler run for certain days or time. You can add multiple rules for various schedules.

You can create exclusion rules for specific schedules or apply a single rule to multiple schedules. You can also enable or disable the rules, or delete the exclusion rules if they are no longer required.

The Run exclusions feature is available when you upgrade to the 10.0 Fix Pack 2 release.

Two new system tables are introduced for this feature. For details about the system tables, see the *IBM Marketing Platform System Tables* guide.

Viewing exclusion rules

Exclusions rules that are already defined for schedules can be viewed from the Run exclusions tab of the Schedule Management page.

The information in the **Previous 1 and next 2 runs** field is shown as per the scheduler definition. It is not currently validated against the exclusion rules.

To view exclusion rules, complete the following steps:

1. Log in to IBM Marketing Platform as the administrator.
2. Click **Settings** > **Schedule management**.
3. Click **Run exclusions**.

You can view the exclusion rules and complete various tasks for the rules. You can also view the status of the rules, the various schedules for which they are applicable, exclusion period, and exclusion type for the rules.

You can also search for exclusion rules by using a wildcard search in the **Filter** text box.

Adding exclusion rules

Exclusion rules can be added for schedules and runs. You can add Absolute or Relative rules, and select the schedules for which the rules will be applicable.

Absolute exclusion rules are set for a set time period. Relative exclusion rules are set only once and was limited to a yearly only. From 11.0 release onwards, in addition to yearly relative date, weekly and monthly date time should be configurable. Exclusion Rules can be enabled or disabled, and can be applied to multiple schedules.

To add an exclusion rule, complete the following steps:

1. Log in to IBM Marketing Platform as the administrator.
2. Click **Settings** > **Schedule management**.
3. On the **Run exclusions** tab, click **Add exclusion rule**.
4. On the **Rule definition** tab, specify the **Rule name**.
5. Optional: Specify the **Description**.
6. Select the **Rule status** as **Enabled** or **Disabled**.

By default, **Enabled** is selected.

7. Select the **Exclusion type**.

   If you select Absolute, complete the following steps:

   a. Select the **Time Zone**.

      By default, the Server default time zone is selected.

   b. Select the **Start date and time**.

   c. Select the **End date and time**.

   If you select **Relative**, complete the following steps:

   a. Select the Time Zone. By default the server time zone is selected.

   b. Select when to start. 1.Now 2.On a date and time - Select a date and time.

   c. Configuring the recurrence pattern to set up relative run Exclusion. Any recurrence pattern you set begins after the start time and End Time you specify. Recurrence pattern options are 1.For weekly user should be able to select one or more day of the week, associated with a start date end time. 2.For monthly user should be able to select a day in a month, associated with a start and end time. 3.For yearly user should be able to select a day in the year, associated with a start and end time.

   d. Select Stop after n occurrences - Relative Run Exclusion Rule stop after the specified number of runs have occurred (whether the Relative Run Exclusion Rule succeed or fail)

   e. Select Stop by a date and time - Relative Run Exclusion Rule are initiated as many times as defined until the specified end date and time is reached.

   Note: A single date of the current year can be selected. Schedules are skipped for the entire day when you select a relateive date.

8. On the **Eligible schedules** tab, select the schedule for which you want to apply the exclusion rule by completing the following steps:

   a. Search for the available schedules by entering a wildcard search in the **Filter** text box.

   b. From **Available schedules**, select the schedules.

   c. Click .

      The selected schedules are moved to the **Selected schedules** table.

   d. Click **Save**.

9. Click **Save**.

Deleting exclusion rules

You can delete the exclusion rules that are available in your system only if the rules are not associated with any schedules or runs.

To delete an exclusion rule, complete the following steps:

1. On the **Run exclusions** tab, select the rule that you want to delete.

   Note: Ensure that the exclusion rule that you want to delete does not have any schedule or run associated with it.

2. Click **Delete**.
3. Confirm the deletion.

Enabling and disabling exclusion rules

You can enable and disable exclusion rules while you create the rules or after you create the rules. By default, a new rule that is created is always Enabled.

When exclusion rules that are applied to schedules are disabled, all schedule runs continue to run as before. When exclusion rules are enabled, the rules are applied to the schedules and schedules are run as per the exclusion criteria that are applied.

To enable or disable an exclusion rule, complete the following steps:

1. On the **Run exclusions** tab, select a disabled rule.
2. Click **Enable**.

   The status of the rule changes to Enabled.
3. To disable a rule, select an enabled rule.
4. Click **Disable**.

   The status of the rule changes to Disabled.

Importing exclusion rules

You can import exclusion rules to apply them to schedules or runs in the system. You can import the rules through an XML file.

The XML file in the specific format must be available to import the exclusion rules. The format of the XML file can be viewed when you click **Import exclusion rules** on the UI.

A sample exclusion rule file is provided with installation and is available in the `<platform_home>\conf\` directory as the `Exclusion_Rule.xml` file.

To import exclusion rules, complete the following steps:

1. On the **Run exclusions** tab, click **Import exclusion rules**.
2. Use the format that is provided to create the XML file to import the rules.
3. Click **Browse** to select the file.
4. Click **Save**.

Understanding the XML file to import exclusion rules

The XML file that can be used to import exclusion rules has certain tags that define the exclusion rules.

Tags in the XML file

The following table lists the tags in the XML file that can be used to import exclusion rules.

Table 25. Tags in the XML file

Tag	Description
ruleName	Name of the exclusion rule.
ruleDescription	Description of the exclusion rule.
ruleStartDate	Date on which the exclusion rule starts. The format of the date must be MM/DD/YYYY.
ruleStartTime	Time at which the exclusion rule starts. The format of the time must be HH:MM:SS.

Table 25. Tags in the XML file (continued)

Tag	Description
ruleEndDate	Date on which the exclusion rule ends. The format of the date must be MM/DD/YYYY.
ruleEndTime	Time at which the exclusion rule ends. The format of the time must be HH:MM:SS.
SchedulerID	IDs of the scheduler on which the exclusion rule must be applied. Multiple scheduler task IDs can be specified. The IDs of the scheduler tasks are available in the USCH_TASK table in the database.
ruleStatus	Status of the exclusion rule. The value can be Enabled or Disabled.

By using the tags, you can define multiple exclusion rules. Reuse the rule tags and modify them as required to define multiple rules.

Example of the XML file to import exclusion rules

An example of the XML file that is used to import exclusion rules is provided for users to reuse the tags and modify the values to create a new XML file according to your requirements.

The following XML tags can be used to create an XML file to import exclusion rules.

```
<rules>
 <rule>
  <ruleName>Rule1</ruleName><!-- specify rule name -->
  <ruleDescription>Rule for skipping 1/13 to 1/19.</ruleDescription><!-- specify rule description -->
  <ruleStartDate>1/13/2017</ruleStartDate><!-- specify exclusion start date. This should be of forma
  <ruleStartTime>8:00:00</ruleStartTime><!-- specify exclusion start time. This should be of format
  <ruleEndDate>1/19/2017</ruleEndDate><!-- specify exclusion end date. This should be of format MM/DD
  <ruleEndTime>18:15:00</ruleEndTime><!-- specify exclusion end time. This should be of format HH:MM
  <SchedulerIDs>
   <SchedulerID>10</SchedulerID> <!-- specify scheduler task Ids, on which this rule should get appl
   <SchedulerID>15</SchedulerID>
  </SchedulerIDs>
  <ruleStatus>Enabled</ruleStatus> <!-- specify exclusion rule status. valid values Enabled/Disabled
 </rule>
</rules>
<rules>
 <rule>
  <ruleName>Rule2</ruleName><!-- specify rule name -->
  <ruleDescription>Rule for skipping 2/6 to 2/10</ruleDescription><!-- specify rule description -->
  <ruleStartDate>2/6/2017</ruleStartDate><!-- specify exclusion start date. This should be of format
  <ruleStartTime>00:00:00</ruleStartTime><!-- specify exclusion start time. This should be of format
  <ruleEndDate>2/10/2017</ruleEndDate><!-- specify exclusion end date. This should be of format MM/DD
  <ruleEndTime>23:59:59</ruleEndTime><!-- specify exclusion end time. This should be of format HH:MM
  <SchedulerIDs>
   <SchedulerID>45</SchedulerID> <!-- specify scheduler task Ids, on which this rule should get appl
   <SchedulerID>88</SchedulerID>
  </SchedulerIDs>
  <ruleStatus>Disabled</ruleStatus> <!-- specify exclusion rule status. valid values Enabled/Disable
 </rule>
</rules>
```

What to consider when you use the scheduler with Campaign

Some special configuration applies when you use the IBM Marketing Software Scheduler with Campaign

- Manual starts of flowchart runs or command-line flowchart commands have no effect on the scheduler, and vice versa with one exception. If a flowchart run is initiated by any means, a subsequent attempt to run the flowchart by any means fails with a lock error if the previous run has not completed.
- Scheduler triggers do not interact in any way with Campaign flowchart triggers. Triggers sent by the Schedule process or by the Campaign trigger utility `unica_actrg` cannot cause schedules in the IBM Marketing Software Scheduler to run, and vice versa.

Difference between the IBM Campaign Schedule process and IBM Marketing Software Scheduler

Starting with the 8.0 release of Marketing Platform, the IBM Marketing Software Scheduler is intended to replace the Campaign Schedule process for scheduling runs of an entire flowchart. The IBM Marketing Software Scheduler is more efficient, as it does not consume any server system resources when the flowchart is not actually running.

The IBM Marketing Software Scheduler starts a flowchart even if it is not running, while the Campaign Schedule process in a flowchart works only if the flowchart is running.

The Campaign Schedule process is preserved for full compatibility with earlier versions, and for other use cases not handled by the IBM Marketing Software Scheduler. For example, you might want to use the Campaign Schedule process to send Campaign triggers or to delay execution of dependent processes.

Do not use the IBM Marketing Software Scheduler to schedule a flowchart that uses the Campaign Schedule process as the top-level process that starts a flowchart run. Typically, only one or the other is necessary. However, if the Schedule process appears in a flowchart that is started by the IBM Marketing Software Scheduler, it functions as configured; conditions required by the IBM Marketing Software Scheduler and the Schedule process must be met before subsequent processes run.

Unlike the IBM Marketing Software Scheduler, the Campaign Schedule process can send external triggers to call command-line scripts. The IBM Marketing Software Scheduler can send triggers only to its own schedules.

Permissions for scheduling flowcharts

Scheduling Campaign flowcharts using the IBM Marketing Software Scheduler requires the following permissions.

Table 26. Permissions for scheduling

Permission	Description
Schedule Batch Flowcharts	Allows scheduling flowcharts using the default run parameters
Schedule Override Batch Flowcharts	Allows overriding the default run parameters for scheduling flowcharts
Run Batch Flowcharts	Allows running flowcharts (required for scheduled flowcharts to run successfully)

Note: When a scheduled flowchart runs, it is run by the Marketing Platform user that created the scheduled task. If this user account is disabled or deleted, any flowcharts previously scheduled by that user will fail to run. If you want to

deactivate this user account but allow these previously scheduled flowcharts to run, leave the user account status set to "active" with only the Run Batch Flowcharts permission granted.

Creating a flowchart schedule using default parameters

Follow this procedure to schedule a flowchart using the default parameters.

1. On a flowchart tab in **View** mode, click the `Run` icon and select **Schedule**.

   The Schedule flowchart dialog box opens.
2. Complete the fields in the Schedule flowchart dialog box.

   If you choose to run more than once, click **Set up Recurrences** to set up a recurrence pattern.
3. Click **Run with this Schedule**.

Important: When you schedule a flowchart, the scheduled task is based on the flowchart name. If the flowchart name is changed after a scheduled task is created, the scheduled task will fail.

About overriding the default parameters for Campaign flowchart run schedules

You can override the default run parameters when you schedule a flowchart run.

When you schedule a Campaign flowchart run, the scheduler uses the default run parameters that have been defined for the flowchart. These parameters include the following:

- The table catalog containing the table mappings that the flowchart uses
- Any user variables values that have been defined within the flowchart
- Login information for any data sources that the flowchart accesses. The default is the user who is scheduling the flowchart.

Campaign allows you override these defaults to run against different data sources or to achieve different results, similar to the capabilities provided by the `unica_svradm` utility. For example, you could schedule multiple runs for a single flowchart to test different combinations of values for user variables. You could specify an alternate table catalog to switch from your production database to a sample database for these test runs. If your organization requires different database logins for test runs and production runs, you can specify the appropriate login information.

Run parameters for scheduling Campaign flowcharts

When you schedule a Campaign flowchart, the flowchart can pass a string containing run parameters to the IBM Marketing Software Scheduler. This string is then passed back to Campaign when a run is started.

In Campaign, all of the values set on the **Override Flowchart Parameters** dialog are passed to the scheduler as a single string. This string is displayed in the **Run parameters** field.

Creating a flowchart schedule

Follow this procedure to schedule a flowchart.

1. On a flowchart tab in **View** mode, click the **Schedules** icon and select **Schedule**.

   The Override flowchart parameters for dialog box opens.

2. If you want to override the default flowchart parameters, complete the fields in the dialog box to specify your flowchart parameters. This step is optional.

   You can add multiple user variables and data sources by clicking the **Add user variable** and **Add data source** links.

   The system does not check syntax of the parameters you enter in these fields. Double-check that you have entered the correct values before proceeding.

   If you do not want to override default flowchart parameters, go on to the next step.
3. Click **Schedule a run** to open the Create a schedule dialog box.

   You can define when the schedule runs and optionally set up recurrences, triggers, and throttling.
4. Click **Run with this schedule**.

Important: When you schedule a flowchart, the scheduled task is based on the flowchart name. If the flowchart name is changed after a scheduled task is created, the scheduled task fails.

The Override Flowchart Parameters page

The following table describes the fields on the Override flowchart parameters dialog. All of the editable fields in this dialog are optional. The system does not check syntax of the parameters you enter in these fields. Double-check that you have entered the correct values before proceeding.

The values you enter in this dialog are shown on the next page of the wizard in the **Run parameters** field.

Table 27. Fields on the Override Flowchart Parameters page

Field	Description
Flowchart Id	Unique ID for the flowchart. This field is filled automatically, and is read-only.
Campaign - Flowchart name	The name of the campaign, campaign code, and flowchart name. This field is filled automatically, and is read-only.
Catalog file name	Specify a stored table catalog file to use for this run.
User variable name	Enter the name of any user variable that has been defined within the flowchart.
Value	Enter a value for the user variable.
Data source name	Enter the name of any data source that the flowchart accesses.
Login	Use this field to override the default login name for the specified data source. The default is the login name of the user who is creating the schedule.
Password	Use this field to override the default password for the specified data source. The default is the password of the user who is creating the schedule.

Schedule notifications

You can set up notifications for any schedule, to alert you to the status of scheduled runs. In addition, users with Administrator permissions in Marketing Platform can set up groups to which notifications are sent.

Individual schedule notifications

You can create notifications for your schedules only after you have created and saved the schedule, not during the creation process. You can configure which statuses trigger a notification, and whether the notifications for each schedule are

sent to your email account, or appear in your notification in-box, or both.

Group schedule notifications

If you want users other than the creator of a schedule to receive schedule notifications, you can enable group-based notifications. You must have administrator permissions in Marketing Platform to set up group notifications.

A configuration property, **Group Name to receive the Job Notifications**, is included for each object type that can be scheduled under the **Platform | Scheduler | Schedule registration | [Product] | [Object type]** category on the **Settings > Configuration** page. All members of the group specified in this configuration property receive notifications for all schedules for that object type (for example, Campaign flowcharts).

Group members receive notifications set up for scheduled runs that have the **Long Duration** or **Not Started/Queued** status. They do not receive notifications for runs with the **On Failure**, **On Success**, or **Unknown/"Other" problem** status.

By adding or removing users in a group, you can control who receives these notifications.

Related concepts:

Chapter 14, “Alert and notification management,” on page 151

Setting up notifications for schedules you create

Use this procedure to set up notifications on schedules you create. You can create notifications only after a schedule has been created and saved, not during the creation process.

1. Select **Settings > Schedule management** and click the name of the schedule for which you want to set up notifications.
2. Click **Edit job notifications** to open the My job notifications window, and then click **New**.
3. Complete the fields and click **Save**.

Deleting or modifying notifications for schedules you create

You can delete or modify any notifications you created.

1. Select **Settings > My job notifications** to open the My job notificaitons window.
2. To delete notifications, select the notifications you want to delete and click **Delete**.
3. To modify notifications, click the name of the notification you want to modify to open the Edit Job Notification window, where you can make and save changes.

Setting up schedule notifications for a group of users

Use this procedure to set up notifications for all schedules that are sent to groups of users that you specify. You must have administrator permissions in Marketing Platform to perform this procedure.

1. On the **Settings > Configuration** page, go to the **IBM Marketing Platform | Scheduler | Schedule registrations** category.

2. For each object type for which you want to enable group-based notifications, set the value of the **Name of group to receive job notifications** property to the name of the group you want to receive notifications for this object type.

   You can use existing groups or create groups for these notifications.

   You might want to set up a group for each object type for which you want to enable group-based notifications.
3. On the User groups page, assign users to the group or groups that you specified in the previous step, as required.

The My job notifications page

You can configure schedule notifications on the My job notifications page.

Table 28. Fields on the My job notifications page

Field	Definitions
Notification title	Enter a name for the notification
Condition	Select the status condition that causes a notification to be sent. You can create a different notification for each status that you want to trigger a notification.
Send the notification to	Select how you want to receive the notification. The notification can be sent to the email account associated with your IBM Marketing Software user account, it can appear in your notifications in the user interface, or both.
Notification status	Select whether this notification is active or inactive. If you select inactive, no notifications are sent.

Schedule management

You can manage all schedules from the **Settings > Schedule management** page. You must have the Administer Scheduled tasks page permission in Marketing Platform to manage schedules.

These are the tabs on the Scheduled tasks page.

- Schedules - On this tab you can create schedules and view or delete schedule definitions. You can click the schedule name to edit a definition, including adding notifications and enabling or disabling the schedule.
- Runs - On this tab you can view queued and completed runs of every schedule, cancel a queued run, and delete a run. You can click the schedule name to edit a definition, including adding notifications and enabling or disabling the schedule.

Schedules and partitions

In a multi-partition environment, you see only the schedules that are created in the partition to which you belong, unless you have the PlatformAdminRole role, which allows you to see all scheduled runs across all partitions.

Unknown status

If you see a large number of runs with a status of Unknown, you can adjust the Scheduler polling frequency by setting the **Platform | Scheduler | Maximum Unknown Status Polling Count** property on the **Settings > Configuration** page. This property specifies the number of times the Scheduler checks the status of a run before reporting a status of Unknown.

The Unknown status indicates that Marketing Platform can not determine whether the job is still running, completed or failed.

If your organization has a large number of scheduled jobs, increasing polling frequency can affect performance.

The schedule list filter

You can filter the schedule list on the Runs and Schedules tabs.

You can enter text in the box at the top right of the list for a quick filter that compares your search term against the values in all of the columns in the list. If your search string is contained in any of the columns, the schedule or run is included in the search result.

For advanced search, you can click **Edit the schedule list filter** to open a window where you can set criteria to evaluate against the attributes of the listed schedules or runs.

Disabling and enabling multiple schedules (with FixPack 10.0.0.1 only)

If you have applied Marketing Platform FixPack 10.0.0.1, you can select multiple schedules on the Schedules tab and disable or enable them by clicking the **Disable** or **Enable** button at the top of the list.

You can use this bulk disable and enable feature in conjunction with the filter to obtain a list of the schedules you want to disable or enable. For example, if you added search tags when you created schedules, you can filter the list to show only schedules with a specific tag. Then you can select all of these schedules and disable or enable them with a single click.

When you disable a scheduled task, any schedules that depend on a trigger from the disabled task are not disabled, but they will not run because they will not receive the trigger.

Related reference:

"The Schedule management pages"

The Schedule management pages

You access the scheduler management pages by selecting **Settings > Schedule Management** or by selecting **View when scheduled** from a flowchart's **Run** menu.

The Schedules tab

Table 29. Fields and links on the Schedules tab

Field or link	Description
Disable	Disable one or more selected schedules. Available only if you have applied Marketing Platform FixPack 10.0.0.1.
Enable	Enable one or more selected schedules. Available only if you have applied Marketing Platform FixPack 10.0.0.1.

Table 29. Fields and links on the Schedules tab (continued)

Field or link	Description
Create a schedule	Click to open a wizard where you can set up a schedule.
Edit the schedule list filter	Click to create an advanced filter for the list.
Delete	Delete one or more selected schedules. You can select schedules by clicking in the column at the left of the schedule. To select all schedules, click at the top of the left side column.
Refresh	Click to refresh the list.
Filter	Click to create a simple filter for the list.
Schedule name	The schedule of which the run is an instance.
Schedule state	Whether the schedule is enabled or disabled.
Scheduled item	The name of the object to be run.
Item type	The type of object to be run.
Created by	The user name of the account that created the schedule.
Start trigger	If the schedule depends on a trigger, the trigger that causes the schedule to run.
Start	Date and time when the first run of this task is scheduled.
Recurrence pattern	A description of the recurrence pattern.
End	Date and time when the last run of this task is scheduled. **Note:** Applies to recurring scheduled tasks only.
Previous 1 and next 2 runs	Date and time of the previous run and next two scheduled runs. **Note:** Applies to recurring scheduled tasks only. The information about previous one and next two scheduled runs is shown as per the scheduler definition. It is not currently validated against the exclusion rules.
Dependencies	If the scheduled object depends on other objects, they are listed here.
On success trigger	The string that is sent if the product reports that a run of this schedule has completed successfully. This field is blank if no on success trigger is specified.
On failure trigger	The string that is sent if the product reports that a run of this schedule has failed. This field is blank if no on failure trigger is specified.

The Runs tab

Table 30. Fields and links on the Runs tab

Field or link	Description
Edit the schedule list filter	Click to create an advanced filter for the list.
Delete	Delete one or more selected schedules. You can select schedules by clicking in the column at the left of the schedule. To select all schedules, click at the top of the left side column.
Mark as cancelled	Cancel one or more selected schedules.
Refresh	Click to refresh the list.
Filter	Click to create a simple filter for the list.
Run id	The identification number assigned to the run in the Marketing Platform system tables.
Schedule name	The name specified for the schedule by its creator.
Scheduled item	The name of the object to be run.
Item type	The type of object to be run.

Table 30. Fields and links on the Runs tab (continued)

Field or link	Description
Start	The date and time when the run started.
Last updated	The date and time when the information for this run was updated.
Execution state	State of the run as defined in the scheduler, as follows. • **Scheduled** - The run has not begun. • **Queued** - The scheduler has started the run, but the IBM Marketing Software product has not begun executing the scheduled run due to throttling constraints. • **Completed**- The run has completed and has returned a status of Failed or Succeeded. • **Cancelled** - A user has cancelled a run by clicking **Mark as Cancelled** on the Scheduled Runs page. If the run was queued when the user marked it as cancelled, it does not execute. If the run was executing, this action does not stop the run, but it is marked as cancelled, and any triggers configured for this run are not sent. Also, runs that depend on the cancelled run do not execute. • **Unknown** - Indicates that Marketing Platform can not determine whether the job is still running, completed or failed.
Run status	Status of the object's run as defined by the product executing the run. If the run sends a status of Cancelled, and the run is later started again and sends any other status to the scheduler, the status is updated in this field.
Details	Information about the run as provided by the product. For example, for a flowchart run, details include the flowchart name and ID, the error if the run fails, and the elapsed time if the run succeeds.

Edit the schedule list filter - Schedules

Table 31. Edit the schedule list filter on the Schedules tab

Column	Description
Filter by search tags / keywords	Select this check box if you want to include search tags or keywords in your filter. The string you enter here is matched with strings entered in the **Search tags / keywords** fields when schedules are created.
Search tags / keywords	Enter the search tags or keywords you want to use in your filter.
Filter on other criteria	Select this check box if you want to include additional criteria in your filter.
Run metadata	Select one of the following options to include in your rule. The options are: • **Schedule name** • **Schedule state** • **Item type** • **Created by** • **Scheduled item**
Condition	Select one of the following options to determine how your rule is evaluated. • **Matches** • **Starts with** • **Ends with** • **Contains**

Table 31. Edit the schedule list filter on the Schedules tab (continued)

Column	Description
Value	Enter or select the value you want to apply to the rule. The options vary depending on the metadata you select for the rule. • **Schedule name** Enter any characters. • **Schedule state** Value options are **Enabled** and **Disabled**. • **Item type** Value options are the various schedule types. • **Created by** Enter any characters. Your value is compared with user login names. • **Scheduled item** Enter any characters. The string you enter here is compared with the text in the **Scheduled item** column.
And / Or	Select one of these operators for each rule you create.

Edit the schedule list filter - Runs

Table 32. Edit the schedule list filter on the Runs tab

Column	Description
Filter based on time	Select this check box if you want to show runs that occurred within a specified time interval.
Time zone	If you select an option other than the server default, the search uses the selected time zone to calculate which schedules fall within the date range you specify.
List runs for the last n instances	For recurring runs, specify how many previous runs to show in the list.
List runs from	Specify a time interval for the runs shown in the list.
Filter on other criteria	Select this check box if you want to include additional criteria in your filter.
Run metadata	Select one of the following options to include in your filter. The options are: • **Schedule name** • **Execution state** • **Run status** • **Scheduled item**
Condition	Select one of the following options to determine how your criteria are evaluated. • **Matches** • **Starts with** • **Ends with** • **Contains**

Table 32. Edit the schedule list filter on the Runs tab (continued)

Column	Description
Value	Enter or select the value you want to apply to the filter. The options vary depending on the metadata you select for the rule. • **Schedule name** Enter any characters. • **Execution state** Value options are: – **Queued** – **Running** – **Completed** – **Unknown** – **Cancelled** • **Run status** Value options are **Succeeded**, **Running**, **Cancelled**,**Failed**, and **Unknown**. • **Scheduled item** Enter any characters. The string you enter here is compared with the text in the **Scheduled item** column.
And / Or	Select one of these operators for each rule you create.

Related concepts:

"Schedule management" on page 81

Related reference:

"The create a schedule wizard" on page 69

Chapter 7. SAML 2.0 based federated authentication

IBM Marketing Platform implements a SAML 2.0 based Identity Provider (IdP) that enables a single sign-on federation among IBM Marketing Software products or between IBM Marketing Software products and third party applications.

A federation is a group of IdPs and applications that works together in a trusted environment and provides services to each other using SAML 2.0 (Security Assertion Markup Language) based standards.

Applications that are members of a federation are called Service Providers (SPs). The IdP server and the SPs can be hosted on premises or on cloud.

A SAML 2.0 federation supports a variety of authentication mechanisms for single sign-on. For example, a user can be authenticated in an SP using that application's authentication mechanism (for example, in-house, OAuth, OpenId, SAML, Kerberos), and then the user can access other SPs using federated single sign-on, provided the applications are part of same federation and the user is mapped appropriately.

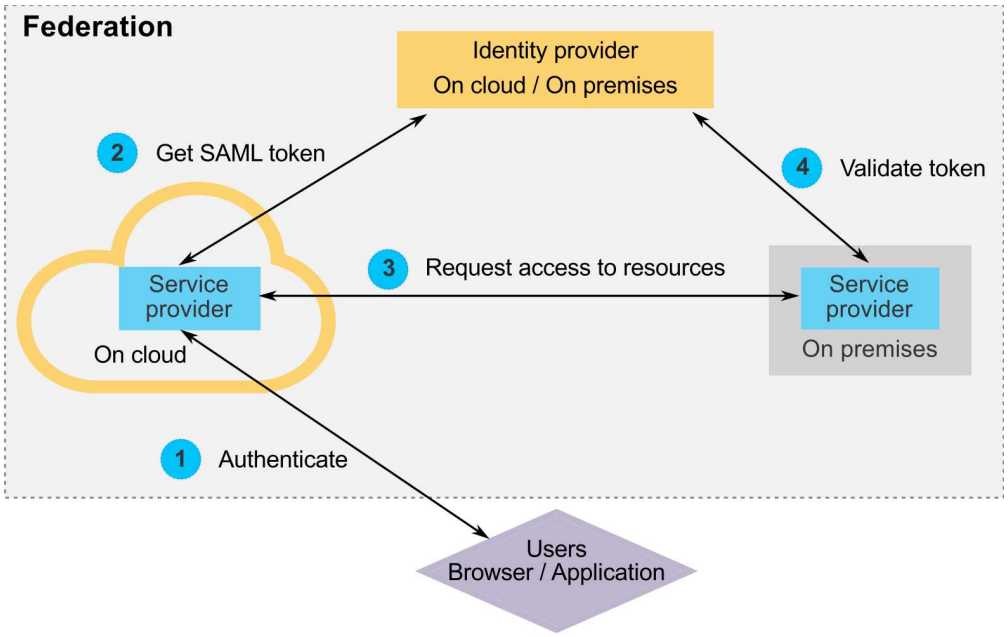
The IdP server creates, validates, or deletes tokens based on user mappings. Data access objects are implemented for the supported database types, and are included in the IdP server.

An administrator maps user IDs between SPs to provide single sign-on access to mapped users. For example, suppose SP_A and SP_B are both members of a federation. User1 is an account in SP_A, and User2 is an account in SP_B. The User1 account is mapped to the User2 account in the federation. When a user logs in to SP_A with User1 credentials, that user has single sign-on access to SP_B. Also, when a user logs in to SP_B with User2 credentials, that user has single sign-on access to SP_A.

© Copyright IBM Corp. 1999, 2018

Diagram

The following diagram illustrates the federation.



Components of the IBM implementation

The IBM implementation of SAML 2.0 based federated single sign-on consists of the following components.

These components are located in the `tools/lib` directory under your Marketing Platform installation.

- A SAML 2.0 based IdP server, delivered as a WAR file: `idp-server.war`
- A client façade: `idp-client.jar`

  The IdP client facade is Java implementation with an API that works with security tokens. It is delivered as a JAR file. Javadoc documentation for the API is included with the Marketing Platform Javadoc.

  The IdP client facade enables Java SPs to quickly integrate with the IdP server and become part of the federation.

Supported use cases

The current implementation enables SPs to work with security tokens to establish single sign-on authentication among the SPs.

Generating a new SAML token

The implementation can generate a new SAML token for a user who initiates a single sign-on authentication request. This user must be mapped on the IdP server. Based on the trusted party's credentials and user mapping, the IdP server creates a new security token and issues it using a SAML 2.0 assertion.

For example, if User1 from SP_A is mapped with User2 from SP_B on the IdP server, and User1 tries to access SP_B resources, the IdP server generates a security token for User1 as a trusted party.

Validating an existing SAML token

The implementation can validate an existing SAML token presented by an SP that receives the access request from a user from another SP. The SP first validates the security token and client mapping with the IdP server to identify the mapped user in its own domain.

For example, when SP_A tries to access SP_B resources on behalf of User1 and presents the IdP security token, SP_B takes this token to the IdP server. If the token is valid and User1 is mapped to an SP_B user, the IdP server resolves the SP_B user in the SP_B domain and returns the assertion.

Deleting an existing SAML token

The implementation can delete an existing SAML token for an SP user when a user logs out from the system or the session times out due to inactivity. Based on the trusted party's credentials and user mapping, the IdP server deletes the token and resets the last accessed timestamp when it receives the logout request. This does NOT delete the user's mapping.

Limitations

The current implementation does not support the following use cases.

- Creating a new user mapping between SP users via a user interface or API
- Updating an existing user mapping between SP users via a user interface or API
- Deleting an existing user mapping between SP users via a user interface or API

Federated authentication and partitions

If your IBM Marketing Software environment has multiple partitions, you can set up separate SAML 2.0 based federated authentication per partition. To implement this, on the **Settings > Configuration** page, you must create a new set of properties in the **IBM Marketing Platform | Security | Federated Authentication | partitions | partition[n]** category for each partition.

Related concepts:

Chapter 16, “Security framework for IBM Marketing Software APIs,” on page 173

How to implement federated authentication

Perform the procedures in this section to implement SAML 2.0 based federated authentication with IBM ExperienceOne products.

Creating the data repository

Create two database tables, `TP_MASTER` and `TP_MAPPING`, to hold user mappings. Any schema can be used to create the tables.

The following example SQL scripts are provided in the `scripts` directory in the `idp-server.war` file.

- DatabaseScript_DB2.sql
- DatabaseScript_Oracle.sql
- DatabaseScript_SQL.sql

The following tables describe the fields in the database tables that the scripts create.

Table 33. Fields in the TP_MASTER table

Field	Description
TP_ID	Primary key. The unique ID for a registered Service Provider.
TP_NAME	The Service Provider name.
TP_INFO	A description of the Service Provider.
KEY_ALIAS	Unique key. The alias name of the Service Provider keystore. Enforces a unique alias name. You can drop the UNIQUE constraint if you want to use the same keystore alias for multiple Service Providers.

Table 34. Fields in the TP_MAPPING table

Field	Description
TP_CLIENT_ID	Foreign key. The TP_ID of the requesting Service Provider. Part of a composite primary key comprised of four columns to ensure that there is no duplicate mapping in this table.
TP_FOR_USER_ID	The ID of the user making the request from the requesting Service Provider. Part of a composite primary key comprised of four columns to ensure that there is no duplicate mapping in this table. Must be a minimum of 4 characters and up to 24 characters long, and contain only alphanumerics, hyphen and underscore: [a-zA-Z0-9_-]
TP_SP_ID	Foreign key. The TP_ID of the serving Service Provider. Part of a composite primary key comprised of four columns to ensure that there is no duplicate mapping in this table. Must be a minimum of 4 characters and up to 24 characters long, and contain only alphanumerics, hyphen and underscore: [a-zA-Z0-9_-]
TP_MAPPED_USER_ID	The ID of the user in the serving Service Provider. Part of a composite primary key comprised of four columns to ensure that there is no duplicate mapping in this table.
SAML_TOKEN	Unique key. ID of the SAML token. Enforces unique token generation. You can drop the UNIQUE constraint if you want to use the same token for multiple Service Providers.
LAST_REQUEST	Timestamp of the last successful request.

Configuring the IdP data source in the web application server

Tomcat, WebSphere®, and WebLogic are supported web application servers for the IdP server. After the IdP server is deployed on the web application server, configure a JNDI data source to connect the IdP server with the data repository.

See the documentation for your web application server for details on how to configure a JNDI data source.

For example, the following configuration is required to create the data source for an Oracle database in a Tomcat server. In the `conf/context.xml` file under your Tomcat installation, define a new resource.

```
<Resource name="idp_datasource"
auth="Container"
type="javax.sql.DataSource"
maxActive="100" maxIdle="30" maxWait="10000"
username="your_username" password="your_password"
driverClassName="com.microsoft.sqlserver.jdbc.SQLServerDriver"
url="jdbc:sqlserver://localhost:1433;DatabaseName=IdPServer"/>
```

Register this resource in the `conf/web.xml` file under your Tomcat installation.

```
<resource-ref>
<description>SQL Server Datasource example</description>
<res-ref-name>idp_datasource</res-ref-name>
<res-type>javax.sql.DataSource</res-type>
<res-auth>Container</res-auth>
</resource-ref>
```

Setting up the classpaths for the IBM IdP client façade

If you want to use the IBM IdP client façade, you must add JAR files in the classpath of your IdP server and the SPs.

1. Obtain the required JAR files as described below, and place these JAR files on your IdP server and the servers that host your SPs.
   - Locate the `unica.war` file in the Marketing Platform installation directory. Extract the `unica.war` file, navigate to the `WEB-INF\lib` directory and copy the following JARs.
     - `bcprov-jdk15.jar`
     - `esapi-2.0.1.jar`
     - `jersey-core-1.17.jar`
     - `jersey-server-1.17.jar`
     - `jersey-servlet-1.17.jar`
     - `joda-time-2.2.jar`
     - `opensaml-2.6.1.jar`
     - `openws-1.5.1.jar`
     - `xmlsec-1.5.6.jar`
     - `xmltooling-1.4.1.jar`
   - `asm-3.1.jar`

     Download from http://mvnrepository.com/artifact/asm/asm/3.1.
   - `jcl-over-slf4j-1.7.5.jar`

     Download from http://mvnrepository.com/artifact/org.slf4j/jcl-over-slf4j/1.7.5.
   - `slf4j-api-1.7.5.jar`

     Download from http://mvnrepository.com/artifact/org.slf4j/slf4j-api/1.7.5.

2. Add the JAR files you obtained in the previous step in the classpath of your IdP server and in the classpath of each of your SPs.
3. For each SP that you want to include in the federation, also add this Client façade JAR file the classpath: : `idp-client.jar` This JAR file is provided with your Marketing Platform installation.

Deploying the IdP server

The `IdP-Server.war` file can be deployed along with the Marketing Platform WAR file in the same server, or it can be deployed separately. There is no direct dependence between these two WAR files.

Configuring the IdP server

The IdP server stores its keystore in its configuration to assert the SAML token coming from SPs. The configurations are stored in the `IdPServerConfig.properties` file under the `conf` folder of the web application server where the IdP server is deployed.

The queries shown in this section are generic. If you need to modify the query for your database type, use one of the following suffixes in the key and enter your new query as the value.

- `Sql`
- `Oracle`
- `db2`

For example, to modify the query in the `com.ibm.ocm.idp.server.query.token.create` property for DB2®, change the property as follows.

```
com.ibm.ocm.idp.server.query.token.create.db2 = new query
```

Note: The sequence and number of columns in your modified query must be the same as in the original query.

Reference: IdPServerConfig.properties file

This section lists the default values of properties in the configuration file, and all supported values for the properties.

`com.ibm.ocm.idp.server.keystore.path`
The absolute path of the keystore file on the web application server host machine.

Default value: *path*/idp.jks

`com.ibm.ocm.idp.server.keystore.passkey`
Passkey of the keystore.

Default value: `idp001`

`com.ibm.ocm.idp.server.keystore.alias`
Alias of the keystore.

Default value: `idp`

`com.ibm.ocm.idp.server.certificate.issuer`
Certificate issuer's URL.

Default value: `http://localhost:8080/idp/`

com.ibm.ocm.idp.server.token.validity
Token validity period in seconds.

Default value: 3600

com.ibm.ocm.idp.server.enable
Logger for IdP server.

Default value: True

com.ibm.ocm.idp.server.dao.class
Database specific data access object implementation.

Supported DAOs are:

com.ibm.ocm.idp.server.dao.IdPServerSQLDAO

com.ibm.ocm.idp.server.dao.IdPServerOracleDAO

com.ibm.ocm.idp.server.dao.IdPServerDB2DAO

Default value: com.ibm.ocm.idp.server.dao.IdPServerSQLDAO

com.ibm.ocm.idp.server.datasource.name
JNDI data source name defined in the application server.

Default value: idp_datasource

com.ibm.ocm.idp.server.query.token.create
Query to create token.

Default value:

```
UPDATE TP_MAPPING
SET SAML_TOKEN = ?, LAST_REQUEST = ?
WHERE TP_CLIENT_ID = ?
AND TP_FOR_USER_ID = ?
AND TP_SP_ID = ?
```

com.ibm.ocm.idp.server.query.token.get
Query to get token.

Default value:

```
SELECT SAML_TOKEN,
LAST_REQUEST FROM TP_MAPPING
WHERE TP_CLIENT_ID = ?
AND TP_FOR_USER_ID = ?
AND TP_SP_ID = ?
```

com.ibm.ocm.idp.server.query.mapping.validate
Query to validate a user mapping.

Default value:

```
SELECT TP_MAPPED_USER_ID FROM TP_MAPPING
WHERE TP_CLIENT_ID = ?
AND TP_FOR_USER_ID = ?
AND TP_SP_ID = ?
```

com.ibm.ocm.idp.server.query.token.delete
Query to delete token.

Default value:

```
UPDATE TP_MAPPING SET SAML_TOKEN = null,
LAST_REQUEST = null
WHERE TP_CLIENT_ID = ?
AND TP_FOR_USER_ID = ?
AND TP_SP_ID = ?
```

com.ibm.ocm.idp.server.query.client.get

Query to get client details.

Default value:

```
SELECT TP_ID, TP_NAME, TP_INFO, KEY_ALIAS
FROM TP_MASTER
WHERE TP_ID = ?
```

Obtaining keystores and importing them into the IdP server

To establish the trusted party assertion, individual keystores are required for each integrating application and the IdP server.

Obtain keystores for the IdP server and for all SPs you want to include in the federation. You can generate the keystores using the Java keytool utility, or you can obtain them from a certificate authority.

If you generate keystores using the keytool utility, here is a typical workflow for this task, with example commands. In the examples, the Java 6 keytool path is `C:\Program Files (x86)\Java\jre7\bin\keytool`.

- The IdP administrator generates a keystore for the IdP server and exports the certificate.

  ```
  # Generate IdP JKS from keytool
  c:\temp> "keytool_path\keytool" -genkey -keyalg RSA -alias idp
  -keystore idp.jks -storepass idp001 -validity 360 -keysize 2048
  # Export IdP certificate from JKS
  c:\temp> "keytool_path\keytool" -export -alias idp -file idp.cer
  -keystore idp.jks
  ```

- An SP administrator generates a keystore and exports it.

  ```
  # Generate Service Provider JKS from keytool
  c:\temp> "keytool_path\keytool" -genkey -keyalg RSA -alias SP_1
  -keystore SP_1.jks -storepass SP001 -validity 360 -keysize 2048
  # Export Service Provider certificate from JKS
  c:\temp> "keytool_path\keytool" -export -alias SP_1 -file SP_1.cer
  -keystore SP_1.jks
  ```

  The SP administrator then sends the certificate to the IdP administrator.

- The IdP administrator imports the SP certificate into the IdP server.

  ```
  # Import Service Provider certificate into IdP JKS
  c:\temp> "keytool_path\keytool" -import -alias SP_1
  -trustcacerts -file SP_1.cer -keystore idp.jks
  ```

Setting configuration properties on the Configuration page

Set configuration properties on the **Settings > Configuration** page to configure federated authentication in IBM Marketing Software.

Set configuration properties under the following categories.

- **IBM Marketing Platform | Security | Federated Authentication**
- **IBM Marketing Platform | Security | Federated Authentication | partitions | partition[n]**

See each property's context help or the related topic links in this section for instructions on setting the values.

Related reference:

"IBM Marketing Platform | Security | Federated authentication" on page 269

"IBM Marketing Platform | Security | Federated authentication | partitions | partition[n]" on page 270

Onboarding Service Providers and users

The IdP server administrator must make one-time entries in the TP_MASTER table to onboard SPs and users.

Here is example SQL for onboarding an SP.

```
INSERT INTO TP_MASTER
 (TP_ID, TP_NAME, TP_INFO, KEY_ALIAS)
 VALUES
 ('SP_Id', 'SP display name', 'SP description', 'keystore alias name')
```

After the trusted parties are registered with the IdP server, the IdP server administrator can map users to participate in federated single sign-on.

The user mapping must be strictly one-to-one between two SPs. For example, User1 from SP_A must be mapped ONLY to any one user in SP_B. However, User1 from SP_A can be mapped with another user in SP_C in the same federation.

Here is an example query for adding users in the TP_MAPPING table.

```
INSERT INTO TP_MAPPING
(TP_CLIENT_ID, TP_FOR_USER_ID, TP_SP_ID, TP_MAPPED_USER_ID, SAML_TOKEN)
VALUES
('SP1_Id', 'SP1_user_Id', 'SP2_Id', 'SP2_user_id','dummy1')
```

Note: The entries for TP_SP_ID and TP_FOR_USER_ID must be a minimum of 4 characters and up to 24 characters long, and contain only alphanumeric, hyphen and underscore characters: [a-zA-Z0-9_-]. Insert unique dummy entries for the SAML_TOKEN column, as this column does not allow nulls and duplicates.

Using the IdP client façade to generate tokens and pass them to Service Providers

When a user is authenticated and wants to access the services of another SP, call the following code on the SP side.

The code generates the federated token.

```
// One time properties to initialize the IdP client.
Properties properties = new Properties();
properties.put(IdPClient.IDP_SERVER_URL, "URL");
properties.put(IdPClient.IDP_CLIENT_CERTIFICATE_ISSUER, "URL");
properties.put(IdPClient.IDP_CLIENT_KEYSTORE_PATH, "JKS file path");
properties.put(IdPClient.IDP_CLIENT_KEYSTORE_PASSKEY, "JKS passkey");
properties.put(IdPClient.IDP_CLIENT_KEYSTORE_ALIAS, "Certificate alias");
// Get the IdP client factory singleton instance
//with the specified parameters.
IdPClientFactory clientFactory = IdPClientFactory.getInstance(properties);
// Get the partition specific client facade to do the assertion.
IdPClientFacade clientFacade = clientFactory.getIdPClientFacade(partition);
// Establish SSO Login with the IdP server
IdPClientToken token = clientFacade.doIdPLogin(clientId, forUserId, spId);
```

After the token is obtained, it can be passed to target SPs to access their resources based on the mapped user's roles and permissions.

```
// Security token is validated at Service Provider side.
IdPClientAssertion assertion = spFacade.assertIdPToken(clientId, forUserId, spId,
token.getTokenId());
// Retrieve the principal from the assertion, if there is no exception.
String principal = assertion.getMappedUser();
```

The client facade is multi-tenant aware and can be used to configure each partition separately. To use this feature, append the client ID to each property name. For example:

```
properties.put(IdPClient.IDP_CLIENT_KEYSTORE_PATH +
".partition1", "JKS file path");
properties.put(IdPClient.IDP_CLIENT_KEYSTORE_PASSKEY +
".partition1", "JKS passkey");
properties.put(IdPClient.IDP_CLIENT_KEYSTORE_ALIAS +
".partition1", "Certificate alias");
```

Reference: RESTful services

Use this information to troubleshoot issues when you use the client facade, or to develop your own SAML 2.0 implementation with the IdP server provided by IBM.

The REST APIs are implemented using an XML data payload. The SAML assertion is directly passed to the POST methods with digital signatures.

Only the POST method is supported for all verbs to ensure unified method access and to enforce security assertions, based on the XML payload. Other methods, such as GET, PUT, and DELETE, return an error message. The following table represents the verbs that implement the supported use cases.

Table 35. Supported verbs

Resource	Post
<idp>/saml/token/clientId/forUserId/spId/create	Generate new SAML token.
<idp>/saml/token/clientId/forUserId/spId/validate	Validate existing SAML token.
<idp>/saml/token/clientId/forUserId/spId/delete	Delete existing SAML token.

Related concepts

This section provides general information about the technologies used in the IBM ExperienceOne implementation of SAML 2.0 based federated single sign-on.

Security Assertion Markup Language 2.0 (SAML 2.0)

SAML 2.0 is a version of the SAML standard for exchanging authentication and authorization data between security domains. SAML 2.0 is an XML-based protocol that uses security tokens containing assertions to pass information about a principal (usually an end user) between a SAML authority, that is, an identity provider, and a SAML consumer, that is, an SP. SAML 2.0 enables web-based authentication and authorization scenarios including cross-domain single sign-on (SSO), which helps reduce the administrative overhead of distributing multiple authentication tokens to the user. For more information, see http://en.wikipedia.org/wiki/SAML_2.0.

Identity Provider (IdP)

Also known as Identity Assertion Provider, the IdP issues identification information for all SPs that interact or provide services within the system. This is achieved via an authentication module that verifies a security token as an alternative to explicitly authenticating a user within a security realm. In perimeter authentication, a user needs to be authenticated only once (single sign-on) and pass along a security token which is processed by an Identity Assertion Provider for each system it needs to access. For more information, see http://en.wikipedia.org/wiki/Identity_provider.

Public-key cryptography

Also known as asymmetric cryptography, a cryptographic algorithm that requires two separate keys, one of which is secret (or private) and one of which is public. Although different, the two parts of this key pair are mathematically linked. The public key is used to encrypt plaintext or to verify a digital signature; whereas the private key is used to decrypt ciphertext or to create a digital signature. For more information, see http://en.wikipedia.org/wiki/Public-key_cryptography.

Chapter 8. SAML 2.0 single sign-on

Marketing Platform supports SAML 2.0 based single sign-on.

In this mode, IBM Marketing Software users can be authenticated against any external or corporate identity provider that follows the standard SAML 2.0 protocol. Identity providers generate the SAML assertion, which is then used by Marketing Platform to allow users to log in. Therefore, a fully functional SAML 2.0 IdP server is a prerequisite for this integration.

After you set up the required configuration properties and a metadata file, users who attempt to log in through the Marketing Platform login page are authenticated through your organization's SAML 2.0 Identity Provider (IdP) server.

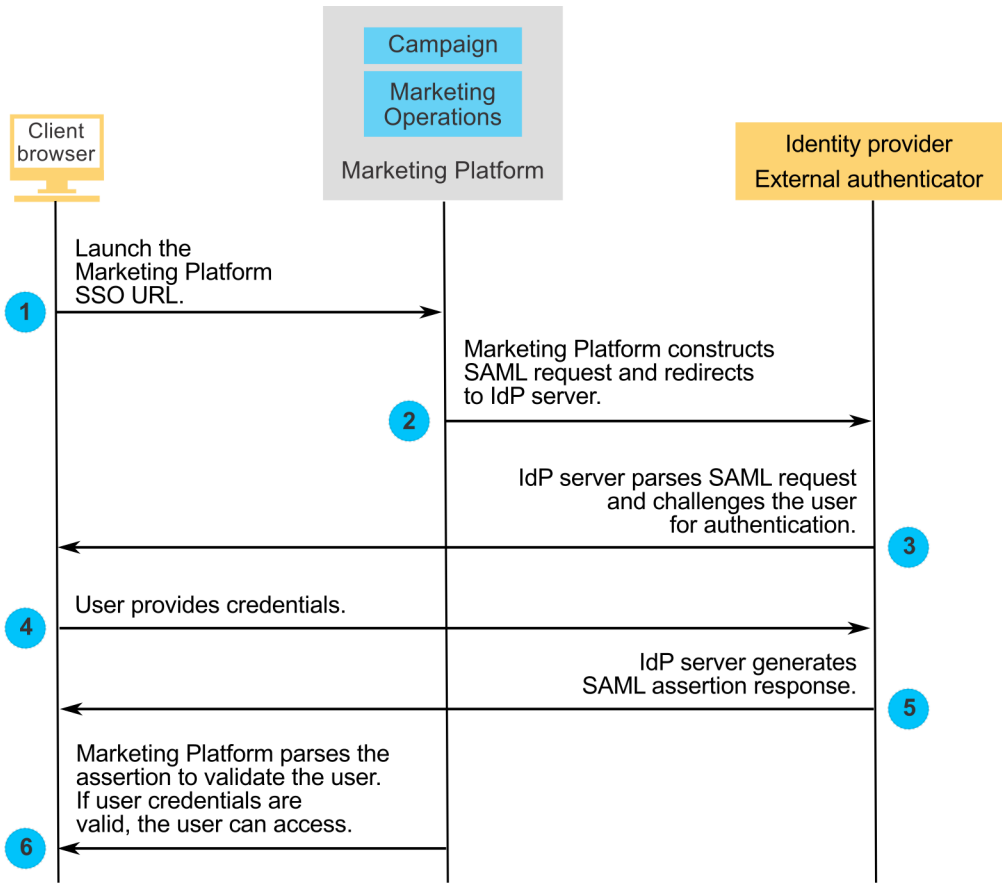
A configuration property, **Add authenticated users to Marketing Platform**, enables automatic creation of a Marketing Platform account for any authenticated user who does not have a Marketing Platform account. These users are automatically added to a default user group, **ExternalUsersGroup**, which has only the **PlatformUser** role initially. Alternatively, you can specify a custom group to which users are added.

If the **Add authenticated users to Marketing Platform** property is not enabled, users must have a Marketing Platform account to log in.

A Marketing Platform administrator can manage group memberships and roles to configure access to IBM Marketing Software products for the automatically created users.

The following diagram illustrates the SAML 2.0 based single sign-on mode in IBM Marketing Software.

© Copyright IBM Corp. 1999, 2018



Behavior when SAML 2.0 single sign-on is implemented

The implementation of SAML 2.0 single sign-on in IBM Marketing Software has the following behavior.

Logout

When logged-in users log out, they are redirected to the standard IBM Marketing Software logout page. A message instructs users to close the browser window to complete the log out process.

Session timeout

If logged-in users are idle for more than 30 minutes they are redirected to the standard IBM Marketing Software logout page. A message instructs users to close the browser window to complete the log out process.

This time out period can be configured in your application server.

Incorrect SAML configuration

If an error occurs due to incorrect SAML configuration when users attempt to log in, the users are redirected to an error page with the message "Login Failed: Bad Credential. Please close the browser window and try again."

User not provisioned in Platform but is a valid IDP user

When the **Add authenticated users to Marketing Platform** property is disabled, users who log in with credentials that are valid for the IdP server but who do not have a Marketing Platform account are redirected to an error page with the message "Login Failed: Bad Credential. Please close the browser window and try again."

User exists in both IDP and Marketing Platform but their Marketing Platform password has expired or been reset

When user's password is expired or reset in Marketing Platform, the user is redirected to an error page with the message "Login Failed: Bad Credential. Please close the browser window and try again."

Configuration process roadmap: SAML 2.0 single sign-on

Use this configuration process roadmap to scan the tasks required to implement SAML 2.0 single sign-on in IBM Marketing Software. Where applicable, the Task column provides links to the topics that describe the tasks in detail.

Table 36. Configuration process roadmap: SAML 2.0 single sign-on

Task	Information
Add Marketing Platform as a service provider in your IdP server. See your IdP server documentation for details.	• Make a note of the application ID assigned to Marketing Platform in your IdP server, as you will use it as the value of a configuration property. • Make a note of password of the key store file, as you will use it as the value of the password in the data source you create.
"Setting up the metadata file"	Set up the XML metadata file generated by your SAML 2.0 IdP server.
"Setting SAML 2.0 configuration properties" on page 102	Set required configuration properties on the **Settings > Configuration page.**
"Setting up a data source for SAML single sign-on" on page 102	Set up the a data source to hold the password of the keystore file.

Setting up the metadata file

Your IdP server generates a metadata file that contains configuration and integration details for SAML 2.0 single sign-on.

Copy the IdP server metadata file and place it on the server where Marketing Platform is installed. Make a note of the following information, which you need when you set configuration property values.

- The location of the metadata file on the Marketing Platform server.
- The value of *entityID* in the XML declaration at the top of the metadata file.

Setting SAML 2.0 configuration properties

To configure SAML 2.0 single sign-on, set properties on the **Configuration > Settings** page.

Set the following properties.

- Set the value of the **Login method** property to **SAML 2.0**.

  This property is located under the **IBM Marketing Platform | Security** node.

  Stop and restart the Marketing Platform web application so that this change takes effect.
- Set properties located under the **IBM Marketing Platform | Security | Login method details | SAML 2.0** node as required.

  See the context help for these properties for details.

Related reference:

"IBM Marketing Platform | Security | Login method details | SAML 2.0" on page 256

Setting up a data source for SAML single sign-on

Save the password of the key store file in a data source in Marketing Platform.

1. Log in to IBM Marketing Software as an Admin user and navigate to the Settings > Users page.
2. Select or create a user and configure a data source for this user as follows.
   - **Data Source** - Enter the value set for the **Key store credential data source** property under **IBM Marketing Platform | Security | Login method details | SAML 2.0** on the **Settings > Configuration** page.
   - **Data Source Login** - Enter the value set for the **Key store credential holder** property under **IBM Marketing Platform | Security | Login method details | SAML 2.0** on the **Settings > Configuration** page.
   - **Data Source Password** - Enter the password of the key store file used for Marketing Platform in your IdP server.

   If you have multiple partitions, you must perform this task in every partition where you have users who should have single sign-on. Alternatively, you can use the platform_admin user account for this step. Because this user is a member of all partitions, the data source is available in all partitions

Sample SAML 2.0 IdP assertion

This section provides examples of the SAML 2.0 request and response.

Example of the SAML 2.0 request generated by Marketing Platform

Marketing Platform generates the SAML 2.0 request shown in this section, and encodes it using OpenSAML Base64 APIs. The request is compatible with any other standard Base64 decoder. The encoded request is posted to the IdP server.

```
<saml2p:AuthnRequest
       xmlns:saml2p="urn:oasis:names:tc:SAML:2.0:protocol"
       AssertionConsumerServiceURL="http://example.com"
       Destination="http://example.com"
       ForceAuthn="false"
```

```
       ID="_0ff13d123291170422ff5e945e9a209e25f3404916451a4aaf"
       IsPassive="false"
       IssueInstant="2015-09-02T14:10:24.376Z"
       ProtocolBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
       Version="2.0">
              <saml2:Issuer xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">
                     IdP_ID
              </saml2:Issuer>
              <saml2p:NameIDPolicy
                     AllowCreate="true"
                     Format="urn:oasis:names:tc:SAML:2.0:nameid-format:transient"
                     SPNameQualifier="SERVICE_PROVIDER_ID"/>
              <saml2p:RequestedAuthnContext
                     Comparison="exact">
                     <saml2:AuthnContextClassRef
                            xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">
                            urn:oasis:names:tc:SAML:2.0:ac:classes:
                            PasswordProtectedTransport
                     </saml2:AuthnContextClassRef>
              </saml2p:RequestedAuthnContext>
</saml2p:AuthnRequest>
```

Example of the SAML 2.0 response generated by the IdP server

```
<samlp:Response
       xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
       Destination="http://serviceprovider.com/location"
              ID="id-wmpfMj-fMh0ihGYJ73rXPTEq7o8-"
       InResponseTo="s2e211c5bfc0200fc48819f381f17d56ca0b5c780f"
       IssueInstant="2015-09-02T14:10:24.376Z"
       Version="2.0">
       <saml:Issuer xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
              Format="urn:oasis:names:tc:SAML:2.0:nameid-format:entity">
              Identity Provider
       </saml:Issuer>
       <samlp:Status>
              <samlp:StatusCode Value="urn:oasis:names:tc:
                     SAML:2.0:status:Success" />
       </samlp:Status>
       <saml:Assertion xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
              ID="idzQO7U5TzPLLL4dlqTqRt9VIOlYg-"
              IssueInstant="2015-09-02T14:10:24.376Z"
              Version="2.0">
              <saml:Issuer Format="urn:oasis:names:tc:SAML:2.0:
                     nameid-format:entity">
                            Identity Provider
              </saml:Issuer>
              <dsig:Signature xmlns:dsig="http://www.w3.org/2000/09/xmldsig#">
                     <dsig:SignedInfo>
                            <dsig:CanonicalizationMethod
                               Algorithm="http://www.w3.org/2001/10/xmlexc-c14n#" />
                            <dsig:SignatureMethod
                               Algorithm="http://www.w3.org/2000/09/xmldsig#
                                          rsa-sha1" />
                            <dsig:Reference URI=
                                   "#id-zQO7U5TzPLLL4dlqTqRt9VIOlYg-" />
                                  <dsig:Transforms>
                                      <dsig:Transform Algorithm=
                                         "http://www.w3.org/2000/09/xmldsig#
                                                 enveloped-signature" />
                                      <dsig:Transform Algorithm=
                                      "http://www.w3.org/2001/10/xml-exc-c14n#" />
                                  </dsig:Transforms>
                                  <dsig:DigestMethod Algorithm=
                                      "http://www.w3.org/2000/09/xmldsig#sha1" />
                                  <dsig:DigestValue>
                                          XXX=
```

```
                                    </dsig:DigestValue>
                                </dsig:Reference>
                        </dsig:SignedInfo>
                        <dsig:SignatureValue>xxx</dsig:SignatureValue>
                </dsig:Signature>
                <saml:Subject>
                        <saml:NameID Format=
                                "urn:oasis:names:tc:SAML:2.0:nameid-format:
                                        transient"
                                NameQualifier="Test Identity Provider"
                                SPNameQualifier="TEST">
                                id-N2EIOvbwaVflUP-cKTzgv8dGYLg-
                        </saml:NameID>
                        <saml:SubjectConfirmation
                                Method="urn:oasis:names:tc:SAML:2.0:cm:bearer">
                                <saml:SubjectConfirmationData
                                  InResponseTo=
                                        "s2e211c5bfc0200fc48819f381f17d56ca0b5c780f"
                                  NotOnOrAfter="2015-09-02T14:10:24.376Z"
                                  Recipient="http://serviceprovider.com/location" />
                        </saml:SubjectConfirmation>
                        </saml:Subject>
                <saml:Conditions
                        NotBefore="2015-09-02T14:10:24.376Z"
                        NotOnOrAfter="2015-09-02T14:10:49.376Z">
                        <saml:AudienceRestriction>
                                <saml:Audience>TEST</saml:Audience>
                        </saml:AudienceRestriction>
                </saml:Conditions>
                <saml:AuthnStatement
                        AuthnInstant="2015-09-02T14:10:24.376Z"
                        SessionIndex="id-1FTYalkjaVTWwHrFRkIRevHfAxk-"
                        SessionNotOnOrAfter="2015-09-02T14:10:38.376Z">
                        <saml:AuthnContext>
                                <saml:AuthnContextClassRef>
                                urn:oasis:names:tc:SAML:2.0:ac:classes:
                                 PasswordProtectedTransport
                                </saml:AuthnContextClassRef>
                        </saml:AuthnContext>
                </saml:AuthnStatement>
                <saml:AttributeStatement xmlns:x500=
                    "urn:oasis:names:tc:SAML:2.0:profiles:
                    attribute:X500"
                    xmlns:xs="http://www.w3.org/2001/XMLSchema"
                    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
                    <saml:Attribute
                           Name="UserIdentifier"
                           NameFormat="urn:oasis:names:tc:SAML:2.0:
                                  attrnameformat:basic">
                          <saml:AttributeValue xsi:type="xs:string">
                                  user@example.com
                           </saml:AttributeValue>
                    </saml:Attribute>
                </saml:AttributeStatement>
        </saml:Assertion>
</samlp:Response>
```

Sample IdP metadata

This section provides an example of the metadata file produced by the IdP server

Example of the metadata file generated by the IdP server

The IdP server generates a metadata file that contains configuration and integration details for SAML 2.0 single sign-on. This file is used by Marketing Platform. Place a copy of the file generated by your IdP server on the Marketing Platform server.

```
<?xml version="1.0" encoding="UTF-8"?>
       <md:EntityDescriptor xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
              entityID="ENTITY_ID">
              <md:IDPSSODescriptor
                     WantAuthnRequestsSigned="false"
                     protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:
                            protocol">
                     <md:KeyDescriptor use="signing">
                            <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
                                   <X509Data>
                                   <X509Certificate>
                                          __certificate__
                                   </X509Certificate>
                                   </X509Data>
                            </KeyInfo>
                     </md:KeyDescriptor>
                     <md:KeyDescriptor  use="encryption">
                            <KeyInfo
                                   xmlns="http://www.w3.org/2000/09/xmldsig#">
                                   <X509Data>
                                          <X509Certificate>
                                                 __certificate__
                                          </X509Certificate>
                                   </X509Data>
                            </KeyInfo>
                            <md:EncryptionMethod
                                   Algorithm=
                                   "http://www.w3.org/2001/04/xmlenc#rsa-1_5"/>
                     </md:KeyDescriptor>
                     <md:ArtifactResolutionService
                            Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
                            Location="__location__" index="0"
                            isDefault="true"/>
                     <md:ArtifactResolutionService
                            Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
                            Location="__location__" index="1"/>
                     <md:SingleLogoutService
                            Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
                            Location="__location__"/>
                     <md:SingleLogoutService Binding=
                            "urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
                            Location="__location__"/>
                     <md:ManageNameIDService
                            Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
                            Location="__location__"/>
                     <md:ManageNameIDService
                            Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
                            Location="__location__"/>
                     <md:NameIDFormat>urn:oasis:names:tc:SAML:2.0:
                            nameid-format:persistent
                     </md:NameIDFormat>
                     <md:NameIDFormat>urn:oasis:names:tc:SAML:2.0:
                            nameid-format:transient
                     </md:NameIDFormat>
                     <md:NameIDFormat>
                            urn:oasis:names:tc:SAML:1.1:
                            nameid-format:emailAddress
                     </md:NameIDFormat>
                     <md:NameIDFormat>urn:oasis:names:tc:SAML:2.0:
                            nameid-format:encrypted</md:NameIDFormat>
```

```
                <md:SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:
                        bindings:HTTP-POST" Location="__location__"/>
        </md:IDPSSODescriptor>
        <md:Organization>
                <md:OrganizationName xml:lang="en">
                        organization_name
                </md:OrganizationName>
                <md:OrganizationDisplayName xml:lang="en">
                        organization_display_name
                </md:OrganizationDisplayName>
                <md:OrganizationURL xml:lang="en"/>
        </md:Organization>
        <md:ContactPerson contactType="technical">
                <md:Company>
                        company
                </md:Company>
                <md:GivenName/>
                <md:SurName/>
                <md:EmailAddress/>
                <md:TelephoneNumber/>
        </md:ContactPerson>
</md:EntityDescriptor>
```

Chapter 9. Configuring JWT authentication between applications

JSON web token (JWT) authentication is used for Journey Designer+Campaign. JWT authentication allows single sign-on between applications.

A request that comes from a calling application contains the JWT token. Marketing Platform validates the request by calling the public key service (PKS). After the JWT token is validated, the request is authenticated and allowed.

This procedure applies only when the 10.0.0.1 FixPack is applied. In version 10.0.0.0, JWT authentication does not use PKS.

Use this procedure to import certificates and set configuration properties to enable JWT authentication.

1. Retrieve the certificate from the public key service (PKS) site.
2. Use the Java keytool to import the certificate into the application server JVM. If your applications are running on different JVMs, import the certificate on each application server JVM.

   For example,

   `/keytool -import -file` *`PKS_Certificate.cer`* `-alias` *`PKS_alias -keystore AppServer_JRE_home`*`/lib/security/cacerts`

   Provide a password. The default keytool password is `changeit`.
3. Set JWT configuration properties on the **Settings > Configuration** page under **IBM Marketing Platform | Security | JWT authentication**.

Related reference:

"IBM Marketing Platform | Security | JWT authentication" on page 272

© Copyright IBM Corp. 1999, 2018

Chapter 10. Single sign-on between IBM Marketing Software and IBM Digital Analytics

If your organization uses IBM Digital Analytics, you can enable single sign-on between Digital Analytics and IBM Marketing Software.

Single sign-on allows users to navigate to Digital Analytics reports from within the IBM Marketing Software user interface without being prompted to log in.

Also, if Digital Analytics reports are referenced in IBM Marketing Software dashboards, single sign-on allows users to view these reports (if they have access to them in Digital Analytics).

Two options for enabling single sign-on between IBM Marketing Software and IBM Digital Analytics

You can choose between two options for enabling single sign-on.

- You can configure Digital Analytics to automatically create an Digital Analytics user account the first time an IBM Marketing Software user navigates to Digital Analytics.

  You might want to choose this option if you want all of your IBM Marketing Software users to have single sign-on with Digital Analytics.
- You can configure IBM Marketing Software user accounts for single sign-on by adding each user's existing Digital Analytics login name to his or her detail page in IBM Marketing Software.

  When you choose this option, users who require access to Digital Analytics must have an Digital Analytics account.

  You might want to choose this option if you want a subset of your IBM Marketing Software users to have single sign-on with Digital Analytics.

Permissions in Digital Analytics for single sign-on users

When the automatic account creation option is **not** selected in Digital Analytics, single sign-on users have the permissions in Digital Analytics that they would have if they log in to Digital Analytics directly.

When the automatic account creation option is selected in Digital Analytics, single sign-on users have permissions in Digital Analytics as follows.

- By default, users have the permissions granted to the Digital Analytics group the administrator has configured for all automatically created users.

  Administrators can modify the permissions associated with this group.
- In addition, the administrator can override automatic account creation for users who already have a Digital Analytics account. If the override is in place for a user, that user has the permissions he or she would have when he or she logs in to Digital Analytics directly.

© Copyright IBM Corp. 1999, 2018

Server clock coordination

The clock on the server on which Marketing Platform is deployed must match the time on the Digital Analytics server clock. For single sign-on, the Digital Analytics server allows for up to 15 minutes of difference (900 seconds) between server clock times.

As a best practice, you should synchronize server clocks. To ensure synchronization, you should use the Network Time Protocol (NTP).

If you cannot synchronize your server clock, and there might be at least 15 minutes of difference between the clocks, you can set the **Clock skew adjustment (seconds)** configuration property under the Coremetrics category in Marketing Platform to a number that reflects the difference between the clocks.

Setting up single sign-on between IBM Marketing Software and Digital Analytics using automatic user account creation

Use this procedure to set up single sign-on between IBM Marketing Software and Digital Analytics using automatic user account creation.

1. Determine the Digital Analytics Client ID you want to use for single sign-on between IBM Marketing Software and Digital Analytics.

   Make a note of the Client ID, as you will need it in a later step.
2. Log in to Digital Analytics as an Admin user with access to the Client ID you selected in the previous step, click the Admin link, and navigate to the Global User Authentication page.
   - In the **IBM Enterprise Marketing Management Shared Secret** field, enter a string that conforms to the rules stated in the instructions next to the field.

     Make a note of this string, as you will need it in a later step.
   - Under Automatic User Account Creation, click **Enabled**.
   - Select a user group to which you want all automatically created users to belong.

     This group should have at least the following Web Analytics permissions.
     - Dashboards > View Standard Dashboards
     - Reports > Site Metrics
     - Reports > Insights
3. Log in to IBM Marketing Software as an Admin user and navigate to the **Settings > Users** page.
4. Select or create a user and configure a data source for this user as follows.
   - **Data Source** - Enter a name.
   - **Data Source Login** - Enter the Client ID you noted in step 1.
   - **Data Source Password** - Enter the Shared Secret you noted in step 2.

   If you have multiple partitions, you must perform this task in every partition where you have users who should have single sign-on.

   Alternatively, you can use the platform_admin user account for this step. Because this user is a member of all partitions, the data source is available in all partitions.
5. In Marketing Platform, navigate to the **Settings > User groups** page and do the following.
   - Create a new group and add the CMUser role to that group.

- Make each user who should have single sign-on a member of that group.

If you have multiple partitions, you must perform this task in every partition where you have users who should have single sign-on.

6. In Marketing Platform, navigate to the **Settings > Configuration** page and set configuration properties as follows.

Table 37. Configuration properties for enabling single sign-on with Digital Analytics

Property	Value
Digital Analytics | Enable IBM Digital Analytics	True
Digital Analytics | Integration | partitions | partition[n] | Platform user for IBM Digital Analytics account	Enter the login name for the Marketing Platform user account that you used in step 4.
Digital Analytics | Integration | partitions | partition[n] | Datasource for IBM Digital Analytics account	Enter the name of the data source you created in step 4.

If you have multiple partitions, you must use the **Digital Analytics \| Integration \| partitions \| partitionTemplate** to create a set of configuration properties for every partition where you have users who should have single sign-on.

The name of the category you create with the template must exactly match the name of the corresponding Campaign partition.

7. For any user for whom you want to override automatic account creation, do the following.
   - In Marketing Platform, navigate to the **Settings > Users** page.
   - Enter the user's Digital Analytics login name in the **Digital Analytics Username** field on the user's detail page.

   This works only for users who already have an Digital Analytics account.

   Note: If an account does not exist in Digital Analytics with this login name, an account will be created for this user with the name you enter here, rather than with the user's Marketing Platform login name.
8. Configure you web application server for single sign-on with Digital Analytics.

Related concepts:

"Digital Analytics configuration properties" on page 277

Related tasks:

"Configuring WebLogic for single sign-on between Digital Analytics and IBM Marketing Software" on page 113

"Configuring WebSphere for single sign-on between Digital Analytics and IBM Marketing Software" on page 113

Setting up single sign-on between IBM Marketing Software and Digital Analytics using manual user account creation

Use this procedure to set up single sign-on between IBM Marketing Software and Digital Analytics using manual user account creation.

1. Determine the Digital Analytics Client ID you want to use for single sign-on between IBM Marketing Software and Digital Analytics.

Make a note of the Client ID, as you will need it in a later step.

2. Log in to Digital Analytics as an Admin user with access to the Client ID you selected in the previous step, click the Admin link, and navigate to the Global User Authentication page.
   - In the **IBM Enterprise Marketing Management Shared Secret** field, enter a string that conforms to the rules stated in the instructions next to the field.

     Make a note of this string, as you will need it in a later step.
   - Under Automatic User Account Creation, click **Disabled**.
3. Log in to IBM Marketing Software as an Admin user and navigate to the **Settings > Users** page.
4. Select or create a user and configure a data source for this user as follows.
   - **Data Source** - Enter a name.
   - **Data Source Login** - Enter the Client ID you noted in step 1.
   - **Data Source Password** - Enter the Shared Secret you noted in step 2.

   If you have multiple partitions, you must perform this task in every partition where you have users who should have single sign-on.

   Alternatively, you can use the platform_admin user account for this step. Because this user is a member of all partitions, the data source is available in all partitions.
5. In Marketing Platform, navigate to the **Settings > User groups** page and do the following.
   - Create a new group and add the DMUser role to that group.
   - Make each user who should have single sign-on a member of that group.

   If you have multiple partitions, you must perform this task in every partition where you have users who should have single sign-on.
6. In Marketing Platform, navigate to the **Settings > Configuration** page and set configuration properties as follows.

Table 38. Configuration properties for enabling single sign-on with Digital Analytics

Property	Value
Digital Analytics | Enable IBM Digital Analytics	True
Digital Analytics | Integration | partitions | partition[n] | Platform user for IBM Digital Analytics account	Enter the login name for Marketing Platform user account that you used in step 4.
Digital Analytics | Integration | partitions | partition[n] | Datasource for IBM Digital Analytics account	Enter the name of the data source you created in step 4.

   If you have multiple partitions, you must use the **Digital Analytics \| Integration \| partitions \| partitionTemplate** to create a set of configuration properties for every partition where you have users who should have single sign-on.

   The name of the category you create with the template must exactly match the name of the corresponding Campaign partition.
7. In Marketing Platform, navigate to the **Settings > Users** page.
8. For each user for whom you want to enable single sign-on, enter that user's Digital Analytics login name in the **IBM Digital Analytics user name** field on the user's Edit properties page.

Note: If a user has exactly the same login names in both IBM Marketing Software and Digital Analytics, you do not have to perform this step.

9. Configure your web application server for single sign-on with Digital Analytics.

Related concepts:

"Digital Analytics configuration properties" on page 277

Related tasks:

"Configuring WebLogic for single sign-on between Digital Analytics and IBM Marketing Software"

"Configuring WebSphere for single sign-on between Digital Analytics and IBM Marketing Software"

Configuring WebLogic for single sign-on between Digital Analytics and IBM Marketing Software

Perform the procedure below in the WebLogic domain where Marketing Platform is deployed to ensure that users can view Digital Analytics reports in dashboards without having to log in.

1. Open the `setDomainEnv` script, located in the `bin` directory under your WebLogic domain directory.
2. Add `-Dweblogic.security.SSL.ignoreHostnameVerification=true` to `JAVA_OPTIONS`.

Related tasks:

"Setting up single sign-on between IBM Marketing Software and Digital Analytics using automatic user account creation" on page 110

"Setting up single sign-on between IBM Marketing Software and Digital Analytics using manual user account creation" on page 111

Configuring WebSphere for single sign-on between Digital Analytics and IBM Marketing Software

Perform the procedure below in WebSphere cell and node where Marketing Platform is deployed to ensure that users can view Digital Analytics reports in dashboards without having to log in.

1. Log in to the WebSphere administrative console.
2. Expand **Security** and click **SSL certificate and key management**.
3. Under **Configuration settings**, click **Manage endpoint security configurations**.
4. Navigate to the outbound configuration for the cell and node where the Marketing Platform is deployed.
5. Under **Related Items**, click **Key stores and certificates** and click the **NodeDefaultTrustStore** key store.
6. Under **Additional Properties**, click **Signer certificates** and **Retrieve From Port**.

   Complete fields as follows.

   - **Host name**: `welcome.coremetrics.com`
   - **Port**: `443`
   - **Alias**: `coremetrics_cert`

Related tasks:

"Setting up single sign-on between IBM Marketing Software and Digital Analytics using automatic user account creation" on page 110

"Setting up single sign-on between IBM Marketing Software and Digital Analytics using manual user account creation" on page 111

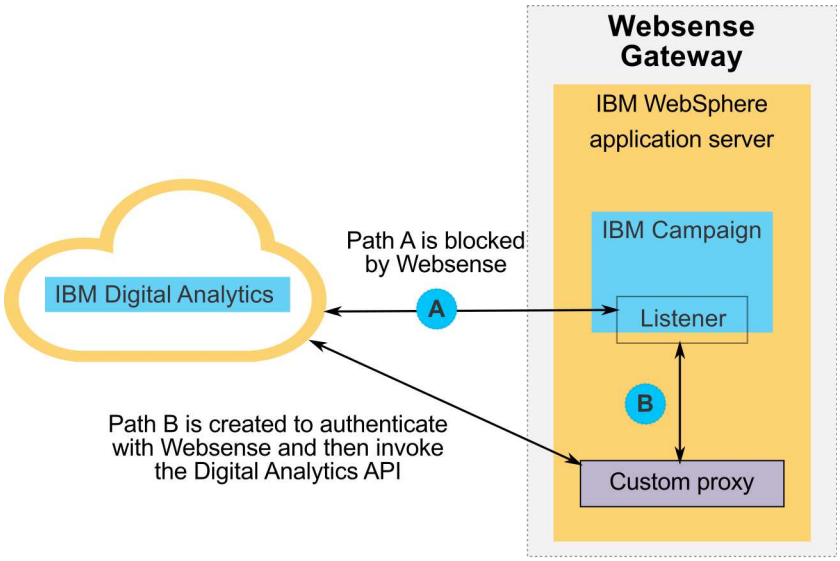
Digital Analytics integration with Websense using a custom proxy

Marketing Platform provides a custom proxy to enable integration between Campaign hosted on premises and Digital Analytics in the cloud when Websense is a required component of the environment.

The custom proxy is supported with the WebSphere application server only.

After the custom proxy is installed, you can configure single sign-on and integration between Digital Analytics and Campaign.

The custom proxy is a Java servlet implementation that acts as a forward proxy. It is injected between the Campaign listener and Digital Analytics. The custom proxy acts as an end point for the Campaign listener to invoke Digital Analytics APIs. Internally, the custom proxy authenticates itself with the Websense content gateway and then securely invokes the APIs outside the network.



Deploying the custom proxy on WebSphere

Perform this procedure to install the custom proxy. This custom proxy is supported with the WebSphere application server only.

Note that you can deploy ProxyServer application in the same WebSphere Profile where you deployed Campaign, or you can use different WebSphere profile.

1. Copy the `ProxyServer.war` file to a location that can be accessed from the WebSphere server.

   You can find the `ProxyServer.war` file in the `tools\lib` directory under your Marketing Platform installation.

2. Deploy the `ProxyServer.war` file, following these guidelines.
   - Select the **Detailed - Show all installation options and parameters** path for installation.
   - You can provide any application name.
   - You do not have to select **Precompile JavaServer Pages files**.
   - On the Initialize parameters for servlets page, complete the fields as shown below.
     - **proxy_host** - Host URL or IP address of the Websense server
     - **proxy_port** - Port number of the Websense server
     - **proxy_username**- User name for Websense authentication
     - **Proxy_password** - Password for Websense authentication
     - **target_url** - End point URL of Digital Analytics, already configured in Campaign
   - On the Map context roots for Web modules page, set the Context Root to `proxy`.
   - When deployment is complete, access the ProxyServer application in a browser at `http://WebSphere_host:Port/proxy`.

     You should receive a message: IBM OCM Secure Proxy Server V.x

Importing the Digital Analytics certificate when WebSphere does not have outbound access

Use this procedure when WebSphere does not have outbound access to the Digital Analytics server.

1. Retrieve the Digital certificate from the Digital Analytics site.

   To retrieve the certificate, go to the Digital Analytics URL and click the lock icon in your browser's address field. Your browser opens a window where you can download the certificate.
2. Import the certificate into the WebSphere JVM using java keytool.

   For example (line breaks added):

   ```
   /keytool -import -file DA_Certificate.cer
   -alias da_alias
   -keystore WebSphere_JRE_home/lib/security/cacerts
   ```

   Provide the password. The default keytool password is changeit.
3. In the WebSphere administrative console, add the following custom properties.
   - javax.net.ssl.trustStore: *WebSphere_JRE_home*/lib/security/cacerts
   - javax.net.ssl.trustStorePassword: *your_password*
   - javax.net.ssl.trustStoreType: jks

Importing the Digital Analytics certificate when WebSphere has outbound access

Use this procedure when WebSphere has outbound access to the Digital Analytics server.

1. In the WebSphere administrative console, expand **Security**, click **SSL certificate and key management**.
2. Under **Configuration settings**, click **Manage endpoint security configurations**.
3. Select the appropriate outbound configuration to navigate to the **(cell):..Node0xCell:(node):..Node0x** management scope.

4. Under **Related Items**, click **Key stores and certificates** and click the **NodeDefaultTrustStore** (or the KeyStore you have used in WebSphere application server) key store.
5. Under **Additional Properties**, click **Signer certificates** and **Retrieve from port**.
   a. In the **Host** field, enter the Digital Analytics server name.
      For example, `export.coremetrics.com`.
   b. In the **Port** field, enter `443`
   c. In the **Alias** field, enter an alias name.
6. Click **Retrieve Signer Information** and verify that the certificate information is for a certificate that you can trust. .
7. Apply and save your configuration.

Next steps

After you install the custom proxy server and import the Digital Analytics certificate, your next steps are to enable single sign-on and configure integration between Digital Analytics and Campaign.

To finish setting up your environment, perform the following procedures.

- Set up single sign on as described in the *IBM Marketing Platform Administrator's Guide, in the chapter titled "Single sign-on between IBM Marketing Software and IBM Digital Analytics."*
- Set up integration as described in the *IBM Campaign Administrator's Guide*, in the chapter titled "IBM Campaign integration with other IBM products."

  Important: The integration procedure includes setting the `ServiceURL` configuration property under **Campaign | partitions | partition[n] | Coremetrics**. When you use the custom proxy, you must set this property to `http://WebSphere_host:Port/proxy`, and restart the Marketing Platform web application.

Related reference:

"Campaign | partitions | partition[n] | Coremetrics" on page 355

Chapter 11. Integration between IBM Marketing Software and Windows Active Directory

Marketing Platform can be configured to integrate with Windows Active Directory server or another LDAP (Lightweight Directory Access Protocol) server. By integrating IBM Marketing Software with a directory server, you can maintain users and groups in one centralized location. Integration provides a flexible model for extending the enterprise authorization policies into IBM Marketing Software applications. Integration reduces support costs and the time needed to deploy an application in production.

See the *Recommended Software Environments and Minimum System Requirements* document for a list of supported directory servers.

Active Directory integration features

Marketing Platform integration with Windows Active Directory provides the features described in this section.

Authentication with Active Directory integration

IBM Marketing Software applications query Marketing Platform for user authorization information.

- Previous versions of Marketing Platform supported NTLMv1 based Microsoft Windows Integrated login. With the arrival of Microsoft Windows 2008 Server and Microsoft Windows 7, the default minimum standard has changed and requires the NTLMv2 protocol. NTLMv2 is not natively supported by Marketing Platform.

  However, you can configure NTLMv2 authentication, so that users are authenticated to all IBM Marketing Software applications when they log in to the corporate network, and no password is required to log in to IBM Marketing Software applications. User authentication is based on their Windows login, bypassing the applications' login screens.

  To configure NTLMv2 authentication, you perform the steps described in this chapter, plus some additional configuration as described in the following developerWorks article:

  https://www.ibm.com/developerworks/community/wikis/home?lang=en#!/wiki/W3a9d2a3c495a_47b6_97a9_e2b8f68b2e88/page/NTLMv2%20authentication%20with%20IBM%20Unica%20Marketing%20Platform.

  This article provides details on how to implement NTLMv2 authentication using Microsoft Internet Information Services (IIS), an application server plug-in for Microsoft Internet Information Services, and a URL re-writer such as ISAPI Rewrite Lite Edition.
- If NTLMv2 authentication is not enabled, users must still log in on the IBM Marketing Software login screen, using their Windows credentials.

Managing internal and external users

When NTLMv2 authentication is enabled, all users are created and maintained in the Active Directory server. (You do not have the option of creating some users in

© Copyright IBM Corp. 1999, 2018

Marketing Platform, which are known as internal users in this guide). If you require the ability to create internal users, do not enable NTLMv2 authentication.

When integration is configured, you cannot add, modify, or delete the imported user accounts in Marketing Platform. You must perform these management tasks on the LDAP side, and your changes are imported when synchronization occurs. If you modify imported user accounts in Marketing Platform, users may encounter problems with authentication.

Any user accounts you delete on the LDAP side are not deleted from Marketing Platform. You should disable these accounts manually in Marketing Platform. It is safer to disable these deleted user accounts rather than deleting them, because users have folder ownership privileges in Campaign, and if you delete a user account that owns a folder, objects in that folder will no longer be available.

Synchronization

When IBM Marketing Software is configured to integrate with an Active Directory server, users and groups are synchronized automatically at pre-defined intervals.

Automatic synchronization has limited functionality.

- Automatic synchronization updates user attributes only. Because group membership changes such as adding, removing, or changing members in a group require administrator oversight, import of these changes is confined to the manual synchronization process by default.
- Users deleted from the LDAP server are not deleted during automatic synchronization.

You can force a full synchronization of all users and groups by using the Synchronize function in the Users area of IBM Marketing Software. Alternatively, you can contact IBM Services to request that they set a hidden configuration property that causes the automatic synchronization to perform a full synchronization.

Importing users based on groups or attributes

You can choose one of two types of filtering to select the user accounts that are imported from the LDAP server into Marketing Platform.

You must choose between group based or attribute based import; multiple methods are not supported simultaneously.

Group based import

Marketing Platform imports groups and their users from the directory server database through a periodic synchronization task that automatically retrieves information from the directory server. When Marketing Platform imports users and groups from the server database, group memberships are not changed. To pick up these changes, you must perform a manual synchronization.

You can assign IBM Marketing Software privileges by mapping an Active Directory group to an IBM Marketing Software group. This mapping allows any new users added to the mapped Active Directory group to assume the privileges set for the corresponding IBM Marketing Software group.

A subgroup in Marketing Platform does not inherit the Active Directory mappings or user memberships assigned to its parents.

Details for configuring group based import are provided in the remainder of this chapter.

Attribute based import

If you do not want to create groups in your Active Directory server that are specific to IBM Marketing Software products, you have the option to control the users who are imported by specifying attributes. To achieve this, you would do the following during the configuration process.

1. Determine the string used in your Active Directory server for the attribute on which you want to filter.
2. Set the **IBM Marketing Platform | Security | LDAP synchronization | LDAP user reference attribute name** property to `DN`.

   This indicates to Marketing Platform that the synchronization is not based on a group with member references but is based on an Org Unit or an Org.
3. When you configure the **LDAP reference map** property, set the Filter portion of the value to the attribute on which you want to search. For the Filter, use the string you determined in step 1.

When you use attribute based synchronization, the periodic synchronization is always a full synchronization, instead of a partial synchronization, which is done for group based synchronization. For attribute based synchronization, you should set the **LDAP sync interval** property to a high value, or set it to `0` to turn off automatic synchronization and rely on manual full synchronization when users are added to the directory.

Follow the instructions provided in the remainder of this chapter to configure integration, using the instructions above in the steps where you set configuration properties.

About Active Directory and partitions

In multi-partition environments, user partition membership is determined by the group to which the user belongs, when that group is assigned to a partition. A user can belong to only one partition. Therefore, if a user is a member of more than one Active Directory group, and these groups are mapped to IBM Marketing Software groups that are assigned to different partitions, the system must choose a single partition for that user.

You should try to avoid this situation. However, if it occurs, the partition of the IBM Marketing Software group most recently mapped to an Active Directory group is the one that the user belongs to. To determine which Active Directory group was most recently mapped, look at the LDAP group mappings displayed in the Configuration area. They are displayed in chronological order, with the most recent mapping listed last.

Special characters in login names

Only three special characters are allowed in login names: dot (.), underscore (_), and hyphen (-). If any other special characters (including spaces) are present in the login name of a user you plan to import into Marketing Platform from your Active Directory server, you must change the login name so that the user does not

encounter issues when logging out or performing administrative tasks (if the user has administration privileges).

Related concepts:

"Configuration process roadmap: Active Directory integration"

Related tasks:

"Forcing synchronization of external users" on page 14

Related reference:

"The user management pages" on page 12

Active Directory integration prerequisites

To take advantage of the Windows Active Directory integration features, IBM Marketing Software applications must be installed on a supported operating system.

In addition, to implement NTLMv2 authentication, users accessing IBM Marketing Software applications must:

- Use a system running a supported Windows operating system.
- Log in as a member of the Windows Active Directory domain against which IBM Marketing Software is authenticating.
- Use a supported browser.

Configuration process roadmap: Active Directory integration

Use this configuration process roadmap to scan the tasks required to integrate IBM Marketing Software with Windows Active Directory. The Topic column provides links to the topics that describe the tasks in detail.

Table 39. Configuration process roadmap: Active Directory integration

Topic	Information
"Obtaining required information" on page 121	Obtain information about your Windows Active Directory server, which is required for integration with IBM Marketing Software.
"Group membership, mapping, and application access" on page 122	If you are using group based synchronization, identify or create the groups in Marketing Platform to which you will map your Active Directory groups.
"Storing directory server credentials in Marketing Platform" on page 123	If your directory server does not allow anonymous access (the most common configuration), configure an IBM Marketing Software user account to hold a directory server administrator user name and password.
• "Setting LDAP login method connection properties in IBM Marketing Software" on page 124 • "Setting LDAP synchronization properties" on page 124 • "Setting user attributes map properties" on page 125 • "Mapping LDAP groups to IBM Marketing Software groups" on page 126	Configure the Marketing Platform for integration by setting values on the Configuration page.
"Testing synchronization" on page 126	Verify that users are imported as expected, and if you are using group based synchronization, verify that users and groups are synchronizing properly.

Table 39. Configuration process roadmap: Active Directory integration (continued)

Topic	Information
"Setting up an Active Directory user with PlatformAdminRole permissions" on page 126	Set up administrator access to Marketing Platform, required when NTLMv2 authentication is enabled.
"Setting the security mode to enable NTLMv2 authentication" on page 127	Set the security mode values on the Configuration page.
Configuring Internet Explorer	Set a custom security level in every instance of Internet Explorer that is used to access IBM Marketing Software. This is required with NTLMv2 authentication, to prevent users from being presented with the IBM Marketing Software login screen.
"Restarting the web application server" on page 127	This step is required to ensure that all of your changes are applied.
"Testing login as an Active Directory user" on page 127	Verify that you can log in to IBM Marketing Software as an Active Directory user.

Obtaining required information

Obtain the required information about the directory server with which you want to integrate. You use this information during the configuration process, to store directory server credentials and to set configuration property values.

Obtain the following information.

- Obtain the server host name and port.
- Identify a user who has search permissions on the directory server, and gather the following information about the user.
  - login name
  - password
  - Distinguished Name (DN)
- Obtain the following for the directory server.
  - Fully qualified host name or IP address
  - The port on which server listens
- Determine the string that your directory server uses for the user attribute in the Group object. Typically, this value is `uniquemember` in LDAP servers and `member` in Windows Active Directory servers. You should verify this on your directory server.
- Obtain the following required user attributes.
  - Determine the string that your directory server uses for the user login attribute. This string is always required. Typically, this value is `uid` in LDAP servers and `sAMAccountName` in Windows Active Directory servers. Verify this string on your directory server.
  - Only if Campaign is installed in a UNIX environment, determine the string that your directory server uses for the alternate login attribute.
- If you are using attribute based synchronization, obtain the strings used for the attributes (one or more) that you want to use for this purpose.
- If you want Marketing Platform to import additional (optional) user attributes stored in your directory server, determine the strings that your directory server uses for the following.
  - First name
  - Last name

- User title
- Department
- Company
- Country
- User email
- Address 1
- Work phone
- Mobile phone
- Home phone

Related concepts:

"About Distinguished Names"

About Distinguished Names

To enable directory server integration in IBM Marketing Software, you must determine the Distinguished Name (DN) for a user and for groups. The DN of an object on the directory server is the complete path through the directory server tree structure to that object.

DNs are made up of these components:

- Organizational Unit (OU). This attribute is used to specify a namespace based on organizational structure. An OU is usually associated with a user-created directory server container or folder.
- Common Name (CN). This attribute represents the object itself within the directory server.
- Domain Component (DC). A Distinguished Name that uses DC attributes has one DC for every domain level below root. In other words, there is a DC attribute for every item separated by a dot in the domain name.

Use your directory server's Administration console to determine an object's Distinguished Name.

Group membership, mapping, and application access

When you plan how to map your directory server groups to Marketing Platform groups, follow the guidelines described here.

- Identify or create the directory server groups whose members you want to import into the Marketing Platform. When these groups are mapped to Marketing Platform groups, members of these groups are automatically created as IBM Marketing Software users.

  Members of your directory server's subgroups are not imported automatically. To import users from subgroups, you must map the subgroups to Marketing Platform groups or subgroups.

  You must map only static directory server groups; dynamic or virtual groups are not supported.
- Identify or create the groups in the Marketing Platform to which you will map directory server groups.
- Assign appropriate application access to the groups you plan to map.

Storing directory server credentials in Marketing Platform

If your directory server does not allow anonymous access, you must configure an IBM Marketing Software user account to hold the user name and password of a directory server user, as described in the following procedure.

1. Log in to IBM Marketing Software as a user with Admin access.
2. Select or create an IBM Marketing Software user account to hold the directory server credentials of an LDAP user with read access over all of the user and group information in the LDAP server. Follow these guidelines.
   - In a later step, you will set the value of the `IBM Marketing Platform user for LDAP credentials` configuration property to the user name for this IBM Marketing Software user account. The default value of this property is `asm_admin`, a user that exists in every new Marketing Platform installation. You can use the `asm_admin`account to hold the directory server credentials.
   - The user name of this IBM Marketing Software user account must not match the user name of any directory server user.
3. Add a data source for this IBM Marketing Software user account to store the credentials that Marketing Platform uses to connect with the LDAP server. Follow these guidelines.

Table 40. Data source fields for storing credentials

Field	Guideline
Data Source Name	You can enter any name, but note that in a later step, the value of the `Data source for LDAP credentials` configuration property must match the data source name you use. To match the default value of this property so that you do not have to set the value, name your data source `LDAPServer`.

Data Source Login	Enter the Distinguished Name (DN) of the administrative user with read access over all of the directory server user and group information that will be synchronized with IBM Marketing Software. The DN resembles the following:
	
	`uidcn=user1,ou=someGroup,dc=systemName,dc=com`
	
	Alternatively, you can use the root user account that has access to all groups on your LDAP server. The default root user and how to specify this user for the supported directory servers are as follows.
	• The root user for Active Directory Server is Administrator. You can specify this user as follows.
	`domain\`*`ldap_admin_username`*
	• The root user for Oracle Directory Server is Directory Manager. You can specify this user as follows.
	`cn=Directory Manager`
	• The root user for IBM Security Directory Server is root. You can specify this user as follows.
	`cn=root`

Data Source Password	Enter the password of the administrative user whose login name you entered in the **Data Source Login** field.

Related tasks:

"Adding internal user data sources" on page 11

Setting LDAP login method connection properties in IBM Marketing Software

LDAP login method properties specify connection details the system uses to connect to the directory server.

1. Click **Settings > Configuration** and navigate to the **IBM Marketing Platform | Security | Login method details | LDAP** category.
2. Set values of the following configuration properties.

   See the related reference for details on how to set each property.

   - LDAP server host name
   - LDAP server port
   - User search filter
   - Use credentials stored in IBM Marketing Platform
   - IBM Marketing Platform user for LDAP credentials
   - Data source for LDAP credentials
   - Base DN
   - Require SSL for LDAP connection

Related reference:

"IBM Marketing Platform | Security | Login method details | LDAP" on page 252

Setting LDAP synchronization properties

LDAP synchronization properties specify details that the system uses to log into the directory server and identify users to import. Some of these properties also control the frequency and other details of the automatic synchronization process.

1. Click **Settings > Configuration** and navigate to the **IBM Marketing Platform | Security | LDAP Synchronization** category.
2. Set values of the following configuration properties in the **LDAP properties** section.

   See each property's context help or the related topic link in this section for instructions on setting the values.

   - LDAP sync enabled
   - LDAP sync interval
   - LDAP sync delay
   - LDAP sync timeout
   - LDAP sync scope
   - LDAP provider URL
   - Require SSL for LDAP connection (optional)
   - LDAP config IBM Marketing Platform group delimiter
   - LDAP reference config delimiter
   - IBM Marketing Platform user for LDAP credentials
   - Data source for LDAP credentials
   - LDAP user reference attribute name
   - LDAP BaseDN periodic search enabled
   - User login
   - Various user attributes such as department, country, and user title (optional)

Related reference:

"IBM Marketing Platform | Security | LDAP synchronization" on page 260

Setting user attributes map properties

These properties specify the user attributes that the system imports from the directory server.

1. Click **Settings > Configuration** and navigate to the **>IBM Marketing Platform | Security | LDAP Synchronization** category.
2. Set values in the **User attributes map** section to map the listed IBM Marketing Software user attributes to the user attributes in your directory server.

   If you are using group based synchronization, the only property you are required to map is `User login`. Typically, this value is `uid` in LDAP servers and `sAMAccountName` in Windows Active Directory servers. Use the value you verified as described in "Obtaining required information."

   If you are using attribute based synchronization, map the attributes on which you want to search.

   Note the following.

   - The properties that you map here are replaced for the imported users each time Marketing Platform synchronizes with your directory server.
   - Marketing Platform requires that email addresses conform to the definition stated in RFC 821. If the email addresses on your directory server do not conform to this standard, do not map them as attributes to be imported.
   - If your directory server database allows an attribute to have more characters than is allowed in the Marketing Platform system tables, as shown in the following table, the attribute value is truncated to fit.

Table 41. Number of characters allowed for user attributes

Attribute	Allowed length
User login (required)	256
First name	128
Last name	128
User title	128
Department	128
Company	128
Country	128
User email	128
Address 1	128
Work phone	20
Mobile phone	20
Home phone	20
Alternate login (required on UNIX)	256

Related reference:

"IBM Marketing Platform | Security | LDAP synchronization" on page 260

Mapping LDAP groups to IBM Marketing Software groups

Users who belong to the directory server groups you map here are imported and made members of the Marketing Platform group or groups specified here.

Important: Do not map any of the groups that have the `asm_admin` user as a member.

1. Click **Settings > Configuration** and navigate to the **IBM Marketing Software | IBM Marketing Platform | Security | LDAP Synchronization | LDAP reference to IBM Marketing Platform group map** category.
2. For each directory server group you want to map to a Marketing Platform group, create an **LDAP reference to IBM Marketing Platform group** category by selecting the *(LDAP reference to IBM Marketing Platform group map)* template. Set the following properties.
   - `New category name`
   - `LDAP reference map`
   - `IBM Marketing Platform group`

   For example, the following values map the LDAP`MarketingPlatformUsers` group to the Marketing Platform `marketingopsUsers` and `campaignUsers` groups (`FILTER` is omitted).
   - `LDAP reference: cn=MarketingPlatformUsers,cn=Users, dc=myCompany,dc=com`
   - `IBM Marketing Platform group: marketingopsUsers;campaignUsers`

Related reference:

"IBM Marketing Platform | Security | LDAP synchronization | LDAP reference to IBM Marketing Platform group map" on page 268

Testing synchronization

Verify that users and groups are correctly synchronized between your servers.

1. Log in to IBM Marketing Software as an IBM Marketing Software user with Admin privileges (not a directory server user).
2. Force synchronization by clicking **Synchronize** on the **Settings > Users** page.
3. Perform the following checks.
   - Verify that users are imported from the LDAP server as expected.
   - If you are using group based synchronization, verify that Marketing Platform group memberships match the expected mapping to directory server groups.

Setting up an Active Directory user with PlatformAdminRole permissions

When NTLMv2 authentication is enabled, you can not log in to IBM Marketing Software as `platform_admin`, so you must perform the following procedure in order to have administrator access to Marketing Platform.

1. Log in to IBM Marketing Software as an internal user (a user created in Marketing Platform rather than a user imported from Active Directory). This must be a user with PlatformAdminRole permissions in the Marketing Platform.
2. Create a Marketing Platform group and assign the PlatformAdminRole role to it.
3. Ensure that at least one Windows Active Directory user is a member of this group.

Related tasks:

"Adding an internal group" on page 21

"Assigning a role to or removing a role from a group" on page 26

"Adding a user to a group or subgroup" on page 22

Setting the security mode to enable NTLMv2 authentication

Only if you want to enable NTLMv2 authentication, set configuration properties as described in this procedure.

To configure NTLMv2 authentication, you must perform the steps described in the following developerWorks article: https://www.ibm.com/developerworks/community/wikis/home?lang=en#!/wiki/W3a9d2a3c495a_47b6_97a9_e2b8f68b2e88/page/NTLMv2%20authentication%20with%20IBM%20Unica%20Marketing%20Platform.

Click **Settings > Configuration** and set configuration properties as shown in the following table.

Table 42. Configuration property values for NTLMv2

Property	Value
IBM Marketing Platform | Security | Login method	Select he `Web access control` option.
IBM Marketing Platform | Security | Login method details | Web access control | Web access control header variable	Enter the variable name as specified in the re-write rules.
IBM Marketing Platform | Security | Login method details | Web access control | Username pattern	Enter `\w*`
General | Navigation | IBM Marketing Platform URL	Enter the IIS site URL.

Related reference:

"IBM Marketing Platform | Security | Login method details | Web access control" on page 255

Restarting the web application server

Restart the web application server to ensure that all of your configuration changes are applied.

Testing login as an Active Directory user

Verify the configuration by logging in to IBM Marketing Software with an appropriate Windows Active Directory user account.

1. Log in to Windows as an Active Directory user who is a member of an Active Directory group mapped to a Marketing Platform group that has been assigned a role in the Marketing Platform.
2. Point your browser to the IBM Marketing Software URL.

   If you have enabled NTLMv2 authentication, you should not see the IBM Marketing Software login screen, and you should be allowed to access the IBM Marketing Software user interface.

If you have not enabled NTLMv2 authentication, you should be able to log in with your Windows credentials.

If you cannot log in, see "restoreAccess" on page 231.

Chapter 12. Integration between IBM Marketing Software and LDAP servers

Marketing Platform can be configured to integrate with Windows Active Directory server or another LDAP (Lightweight Directory Access Protocol) server. By integrating IBM Marketing Software with a directory server, you can maintain users and groups in one centralized location. Integration provides a flexible model for extending the enterprise authorization policies into IBM Marketing Software applications. Integration reduces support costs and the time needed to deploy an application in production.

See the *Recommended Software Environments and Minimum System Requirements* document for a list of supported directory servers.

LDAP integration features

Marketing Platform integration with LDAP provides the features described in this section.

Authentication with LDAP integration

IBM Marketing Software applications query Marketing Platform for user authorization information. When LDAP integration is implemented, users enter their valid LDAP user name and password for authentication to IBM Marketing Software applications.

Managing internal and external users

When integration is configured, you cannot add, modify, or delete the imported user accounts in Marketing Platform. You must perform these management tasks on the LDAP side, and your changes will be imported when synchronization occurs. If you modify imported user accounts in Marketing Platform, users may encounter problems with authentication.

Any user accounts you delete on the LDAP side are not deleted from Marketing Platform. You should disable these accounts manually in Marketing Platform. It is safer to disable these deleted user accounts rather than deleting them, because users have folder ownership privileges in Campaign, and if you delete a user account that owns a folder, objects in that folder will no longer be available.

Synchronization

When IBM Marketing Software is configured to integrate with an LDAP server, users and groups are synchronized automatically at pre-defined intervals.

Automatic synchronization has limited functionality.

- Automatic synchronization updates user attributes only. Because group membership changes such as adding, removing, or changing members in a group require administrator oversight, import of these changes is confined to the manual synchronization process by default.
- Users deleted from the LDAP server are not deleted during automatic synchronization.

© Copyright IBM Corp. 1999, 2018

You can force a full synchronization of all users and groups by using the Synchronize function in the Users area of IBM Marketing Software. Alternatively, you can contact IBM Services to request that they set a hidden configuration property that causes the automatic synchronization to perform a full synchronization.

Importing users based on groups or attributes

You can choose one of two types of filtering to select the user accounts that are imported from the LDAP server into Marketing Platform.

You must choose between group based or attribute based import; multiple methods are not supported simultaneously.

Group based import

Marketing Platform imports groups and their users from the directory server database through a periodic synchronization task that automatically retrieves information from the directory server. When Marketing Platform imports users and groups from the server database, group memberships are not changed. To pick up these changes, you must perform a manual synchronization.

Note: The LDAP groups must have a unique name even if the groups are configured for separate partitions.

You can assign IBM Marketing Software privileges by mapping an LDAP group to an IBM Marketing Software group. This mapping allows any new users added to the mapped LDAP group to assume the privileges set for the corresponding IBM Marketing Software group.

A subgroup in Marketing Platform does not inherit the LDAP mappings or user memberships assigned to its parents.

Details for configuring group based import are provided in the remainder of this chapter.

Attribute based import

If you do not want to create groups in your LDAP server that are specific to IBM Marketing Software products, you have the option to control the users who are imported by specifying attributes. To achieve this, you would do the following during the LDAP configuration process.

1. Determine the string used in your LDAP server for the attribute on which you want to filter.
2. Set the **IBM Marketing Platform | Security | LDAP synchronization | LDAP user reference attribute name** property to `DN`.

   This indicates to Marketing Platform that the synchronization is not based on a group with member references but is based on an Org Unit or an Org.
3. When you configure the **LDAP reference map** property, set the Filter portion of the value to the attribute on which you want to search. For the Filter, use the string you determined in step 1.

When you use attribute based synchronization, the periodic synchronization is always a full synchronization, instead of a partial synchronization, which is done for group based synchronization. For attribute based synchronization, you should

set the **LDAP sync interval** property to a high value, or set it to `0` to turn off automatic synchronization and rely on manual full synchronization when users are added to the directory.

About LDAP and partitions

In multi-partition environments, user partition membership is determined by the group to which the user belongs, when that group is assigned to a partition. A user can belong to only one partition. Therefore, if a user is a member of more than one LDAP group, and these groups are mapped to IBM Marketing Software groups that are assigned to different partitions, the system must choose a single partition for that user.

You should try to avoid this situation. However, if it occurs, the partition of the IBM Marketing Software group most recently mapped to an LDAP group is the one that the user belongs to. To determine which LDAP group was most recently mapped, look at the LDAP group mappings displayed in the Configuration area. They are displayed in chronological order, with the most recent mapping listed last.

Support for internal and external users

IBM Marketing Software supports two types of user accounts and groups.

- **Internal** - User accounts and groups that are created within IBM Marketing Software using the IBM Marketing Software security user interface. These users are authenticated through Marketing Platform.
- **External** - User accounts and groups that are imported into IBM Marketing Software through synchronization with a supported LDAP server. This synchronization occurs only if IBM Marketing Software has been configured to integrate with the LDAP server. These users are authenticated through the LDAP server.

You may want to have both types of users and groups if, for example, you want to give your customers access to IBM Marketing Software applications without adding them to your LDAP server as full corporate users.

Using this hybrid authentication model requires more maintenance than a pure LDAP authentication model does.

Special characters in login names

Only three special characters are allowed in login names: dot (.), underscore (_), and hyphen (-). If any other special characters (including spaces) are present in the login name of a user you plan to import into Marketing Platform from your LDAP server, you must change the login name so that the user does not encounter issues when logging out or performing administrative tasks (if the user has administration privileges).

Related concepts:

"Configuration process roadmap: LDAP integration" on page 132

Related tasks:

"Forcing synchronization of external users" on page 14

Related reference:

"The user management pages" on page 12

LDAP integration prerequisites

To take advantage of the LDAP integration features, IBM Marketing Software applications must be installed on a supported operating system.

Configuration process roadmap: LDAP integration

Use this configuration process roadmap to scan the tasks required to integrate IBM Marketing Software with LDAP. The Topic column provides links to the topics that describe the tasks in detail.

Table 43. Configuration process roadmap: LDAP integration

Topic	Information
"Obtaining required information" on page 121	Obtain information about your LDAP server, which is needed for integration with IBM Marketing Software.
"Group membership, mapping, and application access" on page 122	If you are using group based synchronization, identify or create the groups in Marketing Platform to which you will map your LDAP groups.
"Storing directory server credentials in Marketing Platform" on page 123	If your directory server does not allow anonymous access (the most common configuration), configure an IBM Marketing Software user account to hold a directory server administrator user name and password.
• "Setting LDAP login method connection properties in IBM Marketing Software" on page 124 • "Setting LDAP synchronization properties" on page 124 • "Setting user attributes map properties" on page 125 • "Mapping LDAP groups to IBM Marketing Software groups" on page 126	Configure the Marketing Platform for integration by setting values on the Configuration page.
"Testing synchronization" on page 126	Verify that users are imported as expected, and if you are using group based synchronization, verify that groups are synchronizing properly.
"Setting the security mode to LDAP" on page 138	Set the security mode values on the Configuration page.
"Restarting the web application server" on page 127	This step is required to ensure that all of your changes are applied.
"Testing login as an LDAP user" on page 138	Verify that you can log in to IBM Marketing Software as an LDAP user.

Obtaining required information

Obtain the required information about the directory server with which you want to integrate. You use this information during the configuration process, to store directory server credentials and to set configuration property values.

Obtain the following information.

- Obtain the server host name and port.
- Identify a user who has search permissions on the directory server, and gather the following information about the user.
  - login name
  - password

  - Distinguished Name (DN)
- Obtain the following for the directory server.
  - Fully qualified host name or IP address
  - The port on which server listens
- Determine the string that your directory server uses for the user attribute in the Group object. Typically, this value is `uniquemember` in LDAP servers and `member` in Windows Active Directory servers. You should verify this on your directory server.
- Obtain the following required user attributes.
  - Determine the string that your directory server uses for the user login attribute. This string is always required. Typically, this value is `uid` in LDAP servers and `sAMAccountName` in Windows Active Directory servers. Verify this string on your directory server.
  - Only if Campaign is installed in a UNIX environment, determine the string that your directory server uses for the alternate login attribute.
- If you are using attribute based synchronization, obtain the strings used for the attributes (one or more) that you want to use for this purpose.
- If you want Marketing Platform to import additional (optional) user attributes stored in your directory server, determine the strings that your directory server uses for the following.
  - First name
  - Last name
  - User title
  - Department
  - Company
  - Country
  - User email
  - Address 1
  - Work phone
  - Mobile phone
  - Home phone

Related concepts:

"About Distinguished Names" on page 122

About Distinguished Names

To enable directory server integration in IBM Marketing Software, you must determine the Distinguished Name (DN) for a user and for groups. The DN of an object on the directory server is the complete path through the directory server tree structure to that object.

DNs are made up of these components:

- Organizational Unit (OU). This attribute is used to specify a namespace based on organizational structure. An OU is usually associated with a user-created directory server container or folder.
- Common Name (CN). This attribute represents the object itself within the directory server.

- Domain Component (DC). A Distinguished Name that uses DC attributes has one DC for every domain level below root. In other words, there is a DC attribute for every item separated by a dot in the domain name.

Use your directory server's Administration console to determine an object's Distinguished Name.

Group membership, mapping, and application access

When you plan how to map your directory server groups to Marketing Platform groups, follow the guidelines described here.

- Identify or create the directory server groups whose members you want to import into the Marketing Platform. When these groups are mapped to Marketing Platform groups, members of these groups are automatically created as IBM Marketing Software users.

  Members of your directory server's subgroups are not imported automatically. To import users from subgroups, you must map the subgroups to Marketing Platform groups or subgroups.

  You must map only static directory server groups; dynamic or virtual groups are not supported.
- Identify or create the groups in the Marketing Platform to which you will map directory server groups.
- Assign appropriate application access to the groups you plan to map.

Storing directory server credentials in Marketing Platform

If your directory server does not allow anonymous access, you must configure an IBM Marketing Software user account to hold the user name and password of a directory server user, as described in the following procedure.

1. Log in to IBM Marketing Software as a user with Admin access.
2. Select or create an IBM Marketing Software user account to hold the directory server credentials of an LDAP user with read access over all of the user and group information in the LDAP server. Follow these guidelines.
   - In a later step, you will set the value of the `IBM Marketing Platform user for LDAP credentials` configuration property to the user name for this IBM Marketing Software user account. The default value of this property is `asm_admin`, a user that exists in every new Marketing Platform installation. You can use the `asm_admin`account to hold the directory server credentials.
   - The user name of this IBM Marketing Software user account must not match the user name of any directory server user.
3. Add a data source for this IBM Marketing Software user account to store the credentials that Marketing Platform uses to connect with the LDAP server. Follow these guidelines.

Table 44. Data source fields for storing credentials

Field	Guideline
Data Source Name	You can enter any name, but note that in a later step, the value of the `Data source for LDAP credentials` configuration property must match the data source name you use. To match the default value of this property so that you do not have to set the value, name your data source `LDAPServer`.

Table 44. Data source fields for storing credentials (continued)

Field	Guideline

Data Source Login	Enter the Distinguished Name (DN) of the administrative user with read access over all of the directory server user and group information that will be synchronized with IBM Marketing Software. The DN resembles the following:
	
	`uidcn=user1,ou=someGroup,dc=systemName,dc=com`
	
	Alternatively, you can use the root user account that has access to all groups on your LDAP server. The default root user and how to specify this user for the supported directory servers are as follows.
	• The root user for Active Directory Server is Administrator. You can specify this user as follows.
	`domain\ldap_admin_username`
	• The root user for Oracle Directory Server is Directory Manager. You can specify this user as follows.
	`cn=Directory Manager`
	• The root user for IBM Security Directory Server is root. You can specify this user as follows.
	`cn=root`

Data Source Password	Enter the password of the administrative user whose login name you entered in the **Data Source Login** field.

Related tasks:

"Adding internal user data sources" on page 11

Setting LDAP login method connection properties in IBM Marketing Software

LDAP login method properties specify connection details the system uses to connect to the directory server.

1. Click **Settings > Configuration** and navigate to the **IBM Marketing Platform | Security | Login method details | LDAP** category.
2. Set values of the following configuration properties.

   See the related reference for details on how to set each property.

   - `LDAP server host name`
   - `LDAP server port`
   - `User search filter`
   - `Use credentials stored in IBM Marketing Platform`
   - `IBM Marketing Platform user for LDAP credentials`
   - `Data source for LDAP credentials`
   - `Base DN`
   - `Require SSL for LDAP connection`

Related reference:

"IBM Marketing Platform | Security | Login method details | LDAP" on page 252

Setting LDAP synchronization properties

LDAP synchronization properties specify details that the system uses to log into the directory server and identify users to import. Some of these properties also control the frequency and other details of the automatic synchronization process.

1. Click **Settings > Configuration** and navigate to the **IBM Marketing Platform | Security | LDAP Synchronization** category.
2. Set values of the following configuration properties in the **LDAP properties** section.

   See each property's context help or the related topic link in this section for instructions on setting the values.

   - `LDAP sync enabled`
   - `LDAP sync interval`
   - `LDAP sync delay`
   - `LDAP sync timeout`
   - `LDAP sync scope`
   - `LDAP provider URL`
   - `Require SSL for LDAP connection` (optional)
   - `LDAP config IBM Marketing Platform group delimiter`
   - `LDAP reference config delimiter`
   - `IBM Marketing Platform user for LDAP credentials`
   - `Data source for LDAP credentials`
   - `LDAP user reference attribute name`
   - `LDAP BaseDN periodic search enabled`
   - `User login`
   - Various user attributes such as department, country, and user title (optional)

Related reference:

"IBM Marketing Platform | Security | LDAP synchronization" on page 260

Setting user attributes map properties

These properties specify the user attributes that the system imports from the directory server.

1. Click **Settings > Configuration** and navigate to the **>IBM Marketing Platform | Security | LDAP Synchronization** category.
2. Set values in the **User attributes map** section to map the listed IBM Marketing Software user attributes to the user attributes in your directory server.

   If you are using group based synchronization, the only property you are required to map is `User login`. Typically, this value is `uid` in LDAP servers and `sAMAccountName` in Windows Active Directory servers. Use the value you verified as described in "Obtaining required information."

   If you are using attribute based synchronization, map the attributes on which you want to search.

   Note the following.

   - The properties that you map here are replaced for the imported users each time Marketing Platform synchronizes with your directory server.

- Marketing Platform requires that email addresses conform to the definition stated in RFC 821. If the email addresses on your directory server do not conform to this standard, do not map them as attributes to be imported.
- If your directory server database allows an attribute to have more characters than is allowed in the Marketing Platform system tables, as shown in the following table, the attribute value is truncated to fit.

Table 45. Number of characters allowed for user attributes

Attribute	Allowed length
User login (required)	256
First name	128
Last name	128
User title	128
Department	128
Company	128
Country	128
User email	128
Address 1	128
Work phone	20
Mobile phone	20
Home phone	20
Alternate login (required on UNIX)	256

Related reference:

"IBM Marketing Platform | Security | LDAP synchronization" on page 260

Mapping LDAP groups to IBM Marketing Software groups

Users who belong to the directory server groups you map here are imported and made members of the Marketing Platform group or groups specified here.

Important: Do not map any of the groups that have the `asm_admin` user as a member.

1. Click **Settings > Configuration** and navigate to the **IBM Marketing Software | IBM Marketing Platform | Security | LDAP Synchronization | LDAP reference to IBM Marketing Platform group map** category.
2. For each directory server group you want to map to a Marketing Platform group, create an **LDAP reference to IBM Marketing Platform group** category by selecting the *(LDAP reference to IBM Marketing Platform group map)* template. Set the following properties.
   - `New category name`
   - `LDAP reference map`
   - `IBM Marketing Platform group`

   For example, the following values map the LDAP`MarketingPlatformUsers` group to the Marketing Platform `marketingopsUsers` and `campaignUsers` groups (`FILTER` is omitted).
   - `LDAP reference: cn=MarketingPlatformUsers,cn=Users, dc=myCompany,dc=com`
   - `IBM Marketing Platform group: marketingopsUsers;campaignUsers`

Related reference:

"IBM Marketing Platform | Security | LDAP synchronization | LDAP reference to IBM Marketing Platform group map" on page 268

Testing synchronization

Verify that users and groups are correctly synchronized between your servers.

1. Log in to IBM Marketing Software as an IBM Marketing Software user with Admin privileges (not a directory server user).
2. Force synchronization by clicking **Synchronize** on the **Settings > Users** page.
3. Perform the following checks.
   - Verify that users are imported from the LDAP server as expected.
   - If you are using group based synchronization, verify that Marketing Platform group memberships match the expected mapping to directory server groups.

Setting the security mode to LDAP

Set security mode properties to allow LDAP users to log in to IBM Marketing Software applications.

1. Log in to IBM Marketing Software, click **Settings > Configuration**, and navigate to **IBM Marketing Platform | security**.
2. Set the value of the `Login method` property to `LDAP`.

Restarting the web application server

Restart the web application server to ensure that all of your configuration changes are applied.

Testing login as an LDAP user

Test your configuration by logging in to IBM Marketing Software as an LDAP user who is a member of an LDAP group mapped to a Marketing Platform group that has been assigned access to Marketing Platform.

Chapter 13. Integration with web access control platforms

Organizations use web access control platforms to consolidate their security systems, which provide a portal that regulates user access to web sites. This section provides an overview of IBM Marketing Software integration with web access control platforms.

Authentication

When users access an application through a web access control portal, their authentication is managed through the web access control system. Web access control users who are also members of an LDAP group that is synchronized with IBM Marketing Software are authenticated to all IBM Marketing Software applications when they log in to the web access control system. These users do not see the IBM Marketing Software application login screens.

Authorization

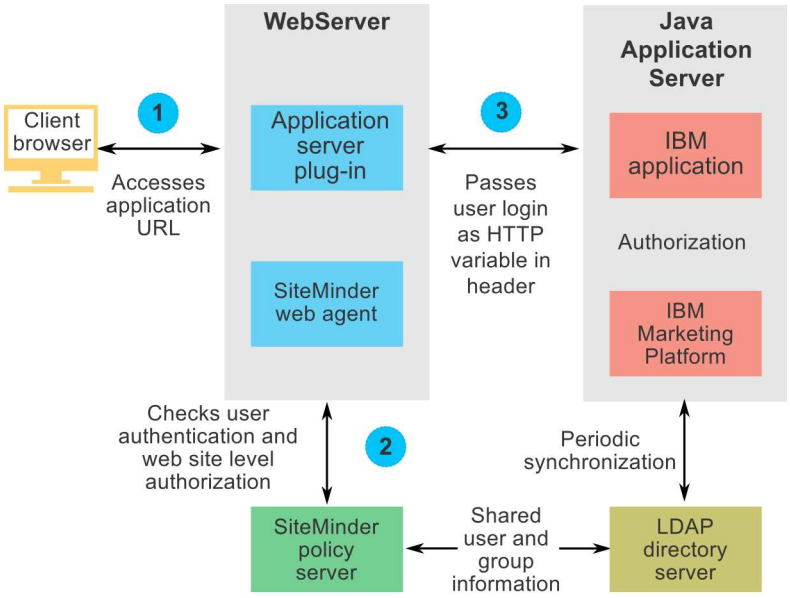
IBM Marketing Software applications query Marketing Platform for user authorization information. Marketing Platform imports groups and their users from the LDAP database through a periodic synchronization task that automatically retrieves information from the LDAP server. When Marketing Platform imports users and groups from the LDAP database, group memberships are maintained. These LDAP users are also exposed to the web access control system, so the web access control system and IBM Marketing Software are referencing a consistent set of users.

Additional authorization controls, including control over the application URLs to which users have access, are also available through most web access control systems.

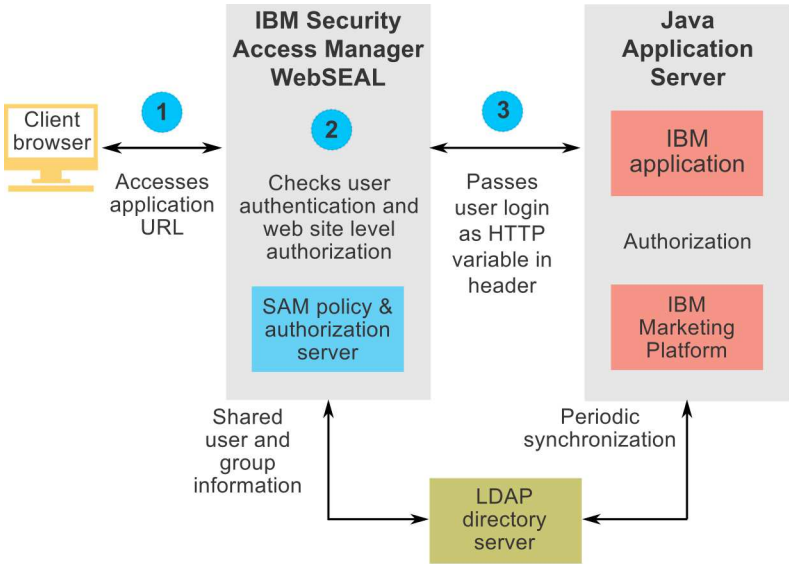
Web access control integration diagrams

The following figure illustrates how IBM Marketing Software works with SiteMinder and an LDAP directory server to authenticate and authorize users.

© Copyright IBM Corp. 1999, 2018



The following figure illustrates how IBM Marketing Software works with IBM Security Access Manager and an LDAP directory server to authenticate and authorize users.



About context roots

You must unprotect URLs in your web access control system to enable various features in IBM Marketing Software products. To perform this task, you need to include the product context roots in the URLs.

The following table provides a list of the default context roots for the IBM Marketing Software products mentioned in this chapter. Your installation might use non-default context roots, but typically most installations accept the default.

The examples in this chapter use the default context roots. If your environment uses a non-standard context root, you must change the context root shown in the example URLs to the context root used in your environment.

Table 46. Context roots for IBM Marketing Software products

Product	Context root
Marketing Platform	unica
Campaign	Campaign
Contact Optimization	Campaign/optimize
Marketing Operations	plan
Distributed Marketing	collaborate
Interact	Campaign/interact

SiteMinder integration prerequisites

The following prerequisites must be met to integrate IBM Marketing Software with Netegrity SiteMinder.

- SiteMinder must be configured to use a web agent and a policy server.
- SiteMinder must be configured to pass the login name as an HTTP variable in the URL request to the IBM Marketing Software application.
- The IBM Marketing Software property **Web access control header variable** must be set to the name of the variable that SiteMinder uses for login names.

  The default name for the SiteMinder login name variable is `sm_user`.
- The SiteMinder policy server must be configured to use LDAP as its repository for storing group members and user properties.
- The IBM Marketing Software application URLs provided by the web server hosting SiteMinder and the Java application server hosting the IBM Marketing Software application must refer to the same path.
- The web server hosting SiteMinder must be configured to redirect requests to the IBM Marketing Software application URL on the Java application server.
- All users who need to access IBM Marketing Software applications must be granted access in SiteMinder to the IBM Marketing Software web applications for HTTP `GET` and `POST` requests through SiteMinder.

See the remainder of this section for settings required to enable specific features or to support certain IBM Marketing Software products.

Configuring SiteMinder for IBM Marketing Software products

Unprotect objects in SiteMinder as described in this procedure to enable correct functioning of your IBM Marketing Software products.

1. Log in to the **Administer Policy Server** area of SiteMinder and click **Domains**.
2. Select the realm that applies to your IBM installations, right-click **unprotecturl**, and select **Properties of Realm**.
3. For each of the applicable URLs as described in the following table, enter the URL in the **Resource Filter** text box, and under **Default Resource Protection**, select **Unprotected**.

Table 47. Un-protected objects required for IBM Marketing Software products

Product or feature	Objects
Campaign	• /Campaign/services/CampaignServices30Service • /Campaign/FlowchartNotifyScheduler • /Campaign/initOfferListResolution.udo • /Campaign/getOfferListResolutionStatus.udo • /Campaign/OperationMonitor • http://*host:port*/Campaign/api/campaign/rest/deepsearch/*partition* Replace *partition* with the partition name. The following apply when integration with Engage is implemented. In the following URLs, replace *partition* with the partition name. • http://*host:port*/Campaign/jsp/engage/engageHome.jsp • http://*host:port*/Campaign/api/campaign/rest/engage/offers • http://*host:port*/Campaign/api/campaign/rest/engage/offer • http://*host:port*/Campaign/servlet/EngageUpload • http://*host:port*/Campaign/api/campaign/rest/engageimportlist/*partition* • http://*host:port*/Campaign/api/campaign/rest/engageimportlist/*partition*/*jobid* This URL is for checking the status of an import job. Replace *jobid* with your job ID. • http://*host:port*/Campaign/api/campaign/rest/engageimportlist/*partition*/schedule • http://*host:port*/Campaign/api/campaign/rest/engageimportlist/*partition*/*channel*/schedule This URL is for sending push or SMS messages. Channel is either sms or push.
Distributed Marketing	• /collaborate/affiniumcollaborate.jsp • /collaborate/services/CollaborateIntegrationServices1.0 • /collaborate/flowchartRunNotifyServlet • /collaborate/js/js_messages.jsp • /collaborate/js/format_symbols.jsp • /collaborate/alertsService
eMessage	/Campaign/emessage/eventSinkServlet
Interact	• /Campaign/interact/saveFlowchartAction.udo • /Campaign/interact/flowchartEventPatterns.udo • /Campaign/interact/testRunFlowchart.udo • /Campaign/interact/getProfileDataAction.udo • /Campaign/interact/manageIPB.udo • /Campaign/initOfferListResolution.udo • /Campaign/getOfferListResolutionStatus.udo

Table 47. Un-protected objects required for IBM Marketing Software products (continued)

Product or feature	Objects
Marketing Operations	• /plan/errorPage.jsp • /plan/alertsService • /plan/services • /plan/services/collabService • /plan/services/PlanIntegrationServices/1.0 • /plan/affiniumplan.jsp • /plan/invalid_user.jsp • /plan/js/js_messages.jsp • /plan/js/format_symbols.jsp • /unica/servlet/AJAXProxy
Contact Optimization	• /Campaign/optimize/ext_runOptimizeSession.do • /Campaign/optimize/ext_optimizeSessionProgress.do • /Campaign/optimize/ext_doLogout.do
IBM SPSS Modeler Advantage Enterprise Marketing Management Edition	/unica/rest/spssUser
Marketing Platform data filters	/unica/servlet/DataFiltering
IBM Marketing Software notifications	• unica/servlet/alertAJAXProxy • unica/notification/alertsCount
IBM Marketing Software Scheduler	/unica/servlet/SchedulerAPIServlet

Related tasks:

"Enabling single logouts with SiteMinder"

Enabling single logouts with SiteMinder

To enable a logout of SiteMinder when a user logs out of an IBM Marketing Software application, configure SiteMinder as follows.

1. Log in to the **Administer Policy Server** area of SiteMinder and set the `logoffUri` property to the URI of the IBM Marketing Software logout page.

   For example: */sm_realm*/unica/j_spring_security_logout where *sm_realm* is the SiteMinder security realm and `unica` is the Marketing Platform context root.
2. Unprotect the IBM Marketing Software logout page, `/unica/jsp/frameworklogout.jsp` to ensure that SiteMinder does not force the user to sign in again to view the logout page.

Related tasks:

"Configuring SiteMinder for IBM Marketing Software products" on page 141

IBM Security Access Manager integration prerequisites

The following prerequisites must be met to integrate IBM Marketing Software with IBM IBM Security Access Manager.

- The IBM Security Access Manager WebSEAL junction must be configured to pass the user name (Short, not Full DN) as the HTTP variable in the URL request to the IBM Marketing Software application.
- The IBM Marketing Software property `Web access control header variable` must be set to the name of the variable that Security Access Manager uses for login names.

  The default name for the Security Access Manager login name variable is `iv-user`.
- The IBM Security Access Manager policy server must be configured to use LDAP as its repository for storing group members and user attributes.
- The IBM Marketing Software application URLs defined by a WebSEAL junction and the Java application server hosting the IBM Marketing Software application must refer to the same path.
- All users who need to access IBM Marketing Software applications must belong to a group added to an Access Control List (ACL) with appropriate permissions. A WebSEAL junction that points to an application server where Marketing Platform is deployed must be attached to this ACL.

Note: When users log out of an IBM Marketing Software application, they are not automatically logged out of IBM Security Access Manager. They must close their browser after they log out of an IBM Marketing Software application to log out of IBM Security Access Manager.

Configuring IBM Security Access Manager for IBM Marketing Software products

Unprotect objects in IBM Security Access Manager as described in this procedure to enable correct functioning of your IBM Marketing Software products.

1. Use Web Portal Manager to log in to the domain as a domain administrator.
2. Click **ACL > Create ACL**, complete the **Name** and **Description** fields, and click **Apply**.
3. Click **ACL > List ACL**, and from the Manage ACLs page, click the link for your ACL policy.
4. From the ACL Properties page, click **Create**, and create two entries for your ACL, as follows.
   - For the first entry, set the entry type to **unauthenticated** and grant **Trx - Traverse, read, and execute** permissions.
   - For the second entry, set the entry type to **Any-other** and grant **Trx - Traverse, read, and execute** permissions.
5. On the ACL Properties page of the ACL, on the Attach tab, attach un-protected objects, as required for your product installations.

   Use the complete path in IBM Security Access Manager, starting from WebSEAL.

Table 48. Un-protected objects required for IBM Marketing Software products

Product or feature	Objects
Campaign	• *WebSEAL junction*/Campaign/services/CampaignServices30Service • *WebSEAL junction*/Campaign/FlowchartNotifyScheduler • *WebSEAL junction*/Campaign/initOfferListResolution.udo • *WebSEAL junction*/Campaign/getOfferListResolutionStatus.udo • *WebSEAL junction*/Campaign/OperationMonitor • http://*host:port*/Campaign/api/campaign/rest/deepsearch/*partition* Replace *partition* with the partition name. The following apply when integration with Engage is implemented. In the following URLs, replace *partition* with the partition name. • http://*host:port*/Campaign/jsp/engage/engageHome.jsp • http://*host:port*/Campaign/api/campaign/rest/engage/offers • http://*host:port*/Campaign/api/campaign/rest/engage/offer • http://*host:port*/Campaign/servlet/EngageUpload • http://*host:port*/Campaign/api/campaign/rest/engageimportlist/*partition* • http://*host:port*/Campaign/api/campaign/rest/engageimportlist/*partition*/*jobid* This URL is for checking the status of an import job. Replace *jobid* with your job ID. • http://*host:port*/Campaign/api/campaign/rest/engageimportlist/*partition*/schedule • http://*host:port*/Campaign/api/campaign/rest/engageimportlist/*partition*/*channel*/schedule This URL is for sending push or SMS messages. Channel is either sms or push.
Distributed Marketing	• *WebSEAL junction*/collaborate/affiniumcollaborate.jsp • *WebSEAL junction*/collaborate/services/CollaborateIntegrationServices1.0 • *WebSEAL junction*/collaborate/flowchartRunNotifyServlet • *WebSEAL junction*/collaborate/js/js_messages.jsp • *WebSEAL junction*/collaborate/js/format_symbols.jsp • *WebSEAL junction*/collaborate/alertsService
eMessage	*WebSEAL junction*/Campaign/emessage/eventSinkServlet

Table 48. Un-protected objects required for IBM Marketing Software products (continued)

Product or feature	Objects
Interact	• *WebSEAL junction*/Campaign/interact/flowchartEventPatterns.udo • *WebSEAL junction*/Campaign/interact/saveFlowchartAction.udo • *WebSEAL junction*/Campaign/interact/testRunFlowchart.udo • *WebSEAL junction*/Campaign/interact/getProfileDataAction.udo • *WebSEAL junction*/Campaign/interact/manageIPB.udo • *WebSEAL junction*/Campaign/initOfferListResolution.udo • *WebSEAL junction*/Campaign/getOfferListResolutionStatus.udo
Marketing Operations	• *WebSEAL junction*/plan/services • *WebSEAL junction*/plan/errorPage.jsp • *WebSEAL junction*/plan/alertsService • *WebSEAL junction*/plan/services/collabService • *WebSEAL junction*/plan/services/PlanIntegrationServices/1.0 • *WebSEAL junction*/plan/affiniumplan.jsp • *WebSEAL junction*/plan/invalid_user.jsp • *WebSEAL junction*/plan/js/js_messages.jsp • *WebSEAL junction*/plan/js/format_symbols.jsp • *WebSEAL junction*/unica/servlet/AJAXProxy
Contact Optimization	• *WebSEAL junction*/Campaign/optimize/ext_runOptimizeSession.do • *WebSEAL junction*/Campaign/optimize/ext_optimizeSessionProgress.do • *WebSEAL junction*/Campaign/optimize/ext_doLogout.do
IBM SPSS Modeler Advantage Enterprise Marketing Management Edition	*WebSEAL junction*/unica/rest/spssUser
Marketing Platform data filters	*WebSEAL junction*/unica/servlet/DataFiltering.
IBM Marketing Software notifications	• *WebSEAL junction*/unica/servlet/DataFiltering • *WebSEAL junction*/unica/servlet/alertAJAXProxy • *WebSEAL junction*/unica/notification/alertsCount
IBM Marketing Software Scheduler	*WebSEAL junction*/unica/servlet/SchedulerAPIServlet
Enable a logout of IBM Security Access Manager when a user logs out of an IBM Marketing Software application	• *WebSEAL junction*/unica/j_spring_security_logout • *WebSEAL junction*/unica/jsp/frameworklogout.jsp

Configuration process roadmap: integrating IBM Marketing Software with a web access control system

Use this configuration process roadmap to scan the tasks required to integrate IBM Marketing Software with a web access control system. The Topic column provides links to the topics that describe the tasks in detail.

Table 49. Configuration process roadmap: integrating IBM Marketing Software with a web access control system

Topic	Information
"Performing LDAP integration"	Follow instructions for LDAP integration, stopping at the "Test synchronization" step.
"Setting web access control connection properties in IBM Marketing Software"	Set web access control integration properties on the Configuration page.
"Restarting the web application server" on page 127	This step is required to ensure that all of your changes are applied.
"Testing web access control synchronization and IBM Marketing Software login" on page 148	Verify that users and groups synchronize correctly in your web access control system and that you can log in to IBM Marketing Software.

Performing LDAP integration

Perform all of the steps required for LDAP integration.

Setting web access control connection properties in IBM Marketing Software

To configure web access control integration, you set some configuration properties.

On the **Settings & Configuration** page, set values of the properties as described in the following table.
See the related reference for details on how to set each property.

Table 50. Properties for configuring web access control integration

Property	Value
IBM Marketing Software | IBM Marketing Platform | Security | Login method details	Select `Web access control`.
IBM Marketing Software | IBM Marketing Platform | Security | Login method details | Web access control | Username pattern	A Java regular expression used to extract the user login from the HTTP header variable in web access control software. You must XML-escape any XML characters in the regular expression. The recommended value for SiteMinder and IBM Security Access Manager is `\w*`
IBM Marketing Software | IBM Marketing Platform | Security | Login method details | Web access control | Web access control header variable	The HTTP header variable configured in the web access control software, which is submitted to the web application server. By default, SiteMinder uses `sm_user`, and IBM Security Access Manager uses `iv-user`. For IBM Security Access Manager, set this value to the user name component of the IBM Raw string, not the IBM HTTP string.

Table 50. Properties for configuring web access control integration (continued)

Property	Value
IBM Marketing Software | General | Navigation | IBM Marketing Platform URL	Set to `http://sm_host:sm_port/sm_realm/unica` where • *sm_host* is the name of the machine on which SiteMinder is installed • *sm_port* is the SiteMinder port number • *sm_realm* is the SiteMinder realm

Related reference:

"IBM Marketing Platform | Security | Login method details | Web access control" on page 255

"IBM Marketing Software | General | Navigation" on page 241

Restarting the web application server

Restart the web application server to ensure that all of your configuration changes are applied.

Testing web access control synchronization and IBM Marketing Software login

Follow this procedure to test your integration.

1. Log in to your web access control system with an LDAP account that has been synchronized into your web access control system and has access to Marketing Platform.
2. Verify that:
   - Users are imported as expected
   - Groups are imported as expected
   - IBM Marketing Software group memberships match the expected mapping to LDAP groups
3. Point your browser to the Marketing Platform URL and log in.

   You should be able to access IBM Marketing Software without being presented with the IBM Marketing Software login screen.
4. Use the following guidelines to resolve problems when your web access control software is Netegrity SiteMinder.
   - If you see an IBM Marketing Software login screen, the user account with which you logged in might not have been synchronized into SiteMinder.
   - If you are not able to access IBM Marketing Software, check that your SiteMinder configuration is correct. You can use the SiteMinder TestTool to verify that the user account with which you logged in has been authorized and granted access to IBM Marketing Software URLs in SiteMinder.
   - If you can access IBM Marketing Software, but navigation is not working correctly or images are not displaying, check to be sure that the web server hosting SiteMinder and the Java application server hosting Marketing Platform use the same path to refer to Marketing Platform.

Configuring integration with an SSL type of WebSEAL junction

Follow this procedure to configure Marketing Platform integration with IBM Security Access Manager using an SSL type of WebSEAL junction.

For details on these procedures, see the documentation provided with IBM Security Access Manager and your web application server.

1. Generate or purchase SSL certificates and configure your web application server to use them.
2. Create a webSEAL certificate and configure IBM Security Access Manager to use it.
3. Import your webSEAL certificate into your web application server.
4. Import your web application server certificate into IBM Security Access Manager.
5. Create an SSL type of WebSEAL junction in IBM Security Access Manager.

   If you install multiple IBM Marketing Software products, create a separate junction for each product.
6. Set the navigation URL configuration property on the **Settings & Configuration** page for each installed product.

   The value should reflect the webSEAL junction used for that product. Follow this pattern:

   `https://`*`machine_name_or_IP_address.domain_name:port_number/webSEAL_junction/context-root`*

   To access IBM Marketing Software, use a URL such as the following:

   `https://`*`machine_name_or_IP_address.domain_name:port_number/webSEAL_junction`*`//unica`
7. Unprotect URLs in IBM Security Access Manager as described elsewhere in this guide.

Related concepts:

Chapter 15, “Implementation of one-way SSL,” on page 153

Chapter 14. Alert and notification management

IBM Marketing Platform provides support for system alerts and user notifications sent by IBM Marketing Software products.

System alerts and user notifications sent by products appear in the user interface, as follows.

- **Alerts** contain information about system events. They appear in a pop-up window when a user logs in.

  Examples are planned or unplanned server shutdowns.
- **Notifications** contain user-specific information about changes made to items in which the user has an interest, or tasks the user must perform. The user can view them by clicking the envelope icon in the top right of the window.

  Examples are updates to a flowchart or mailing list, or reminders about a deadline for an assigned task.

Users can also subscribe to receive alerts and notifications by email, if Marketing Platform has been configured to send them.

Within Marketing Platform, the IBM Marketing Software Scheduler uses the notification feature.

Related concepts:

"Schedule notifications" on page 79

Related tasks:

"Configuring email notifications in IBM Marketing Software" on page 152

Alert and notification subscriptions

Users can choose to have system alerts and notifications delivered in emails, if Marketing Platform is configured to send them. They can also select the level to which they subscribe.

For example, they can choose to receive only Critical system alerts, and receive all notifications. The subscription levels differ depending on the product that is sending the system alerts and notifications.

Note: All system alerts are always delivered in pop-up windows when users log in to IBM Marketing Software. Users cannot control these by changing their subscriptions.

When users log in to IBM Marketing Software, the System alerts window is displayed only if there are any new or unread alerts. Users can mark an alert as read by selecting the alert and clicking **Mark as read** in the System alerts window.

Setting system alert and notification subscriptions

Non-administrative users can set their own subscriptions for system alerts and notifications by following this procedure

1. Log in to IBM Marketing Software and select `Settings > Users`.

© Copyright IBM Corp. 1999, 2018

Your account detail page opens.

2. Click **Notification Subscription** on your account detail page.
3. Use the checkboxes to select the level of notifications you want to receive, and whether to receive them in the user interface, by email, in both places, or not at all.
4. Click **Submit** to save your changes.

Configuring email notifications in IBM Marketing Software

Follow this procedure to configure the Marketing Platform to send system alert and notification emails to users. You must have an email server set up before you start.

Obtain the following information about your mail server.

- The protocol used by your mail server.
- The port on which the mail server listens.
- The name of the machine that hosts your mail server.
- Whether your mail server requires authentication.
- If your mail server requires authentication, an account name and password on the mail server.

Tip: See the related references if you need additional details about performing this procedure.

1. If your mail server requires authentication, save a mail server account name and password as a data source in a Marketing Platform user account.

   Use an internal Marketing Platform user account, not a user imported from an LDAP server.

   Make a note of the Marketing Platform user name and the data source name, as you will use them in step 3.
2. Log in to IBM Marketing Software as a user with administrative privileges in Marketing Platform.
3. On the **Settings > Configuration** page, set the configuration properties in the following categories.
   - `General | Communication | Email`
   - `Platform | Notifications`

   Use the information you obtained about your mail server to set values.

Related concepts:

Chapter 14, “Alert and notification management,” on page 151

Related tasks:

“Adding internal user data sources” on page 11

Related reference:

“IBM Marketing Software | General | Communication | Email” on page 246

“IBM Marketing Platform | Notifications” on page 273

Chapter 15. Implementation of one-way SSL

This section describes one-way SSL in IBM Marketing Software.

Any communication that needs to be secured between two applications connecting over a network can be transmitted using the Secure Sockets Layer (SSL) protocol.

SSL provides secure connections by:

- Allowing an application to authenticate the identity of another application
- Using a private key to encrypt and decrypt data transferred over the SSL connection

When applications are configured for SSL, web traffic is over HTTPS instead of HTTP as reflected in the URLs.

When processes communicate with each other, the process making a request acts as the client and the process responding to a request acts as the server. For complete security, SSL should be implemented for all forms of communication with IBM Marketing Software products.

SSL can be configured one-way or two-way. With one-way SSL, the server is required to present a certificate to the client but the client is not required to present a certificate to the server. To successfully negotiate the SSL connection, the client must authenticate the server. The server accepts a connection from any client.

Related tasks:

"Configuring integration with an SSL type of WebSEAL junction" on page 149

Overview of SSL certificates

Read this section to understand SSL certificates in general.

What is a certificate?

A certificate is a digital signature that identifies the server as some named entity. Certificates can be signed by a certificate authority (CA) that vouches for the identity of the server, or they can be self-signed. Verisign or Thawte are examples of CAs. A self-signed certificate is one where the CA is the same entity that the certificate claims to identify.

Server-side certificates

Every server that is intended to provide SSL communication, whether it is an application server or an IBM Marketing Software application such as the Campaign listener, needs to serve up a certificate.

© Copyright IBM Corp. 1999, 2018

Client side truststores

When the client receives the server certificate, it is up to the client to determine whether to trust the certificate. A client trusts a server certificate automatically if the certificate exists in the client truststore. A truststore is a database of trusted certificates.

Modern browsers have a truststore loaded with the common certificates endorsed by CAs. This is why you are not prompted when entering the secured site at major merchant web sites - they use certificates signed by a CA. But, when you log in to an IBM application that serves up a self-signed certificate, you sees the prompt.

Browsers check that the host name of the server matches the subject name in the certificate (the subject name is the Common Name used in the Distinguished Name, which you supply when you request a certificate). The browser might issue a warning if these two names do not match.

When a browser accesses an IBM application secured with a certificate it does not recognize (for example, a self-signed certificate), a dialog window opens, asking if the user wants to continue. If the user chooses to install the certificate to the local truststore, the prompt does not appear again.

Client and server roles in IBM Marketing Software

IBM Marketing Software application components can act as either the client or the server in a communication, depending on the situation.

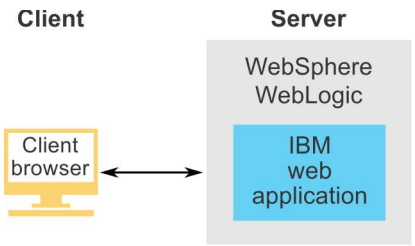
Most IBM Marketing Software applications consist of two parts.

- The web application. The web application is the component that users access through a browser.
- The server (for example, the Campaign listener and the Marketing Platform API server). This component is accessed programmatically.

The following examples and diagrams illustrate the roles played by IBM components in various communications.

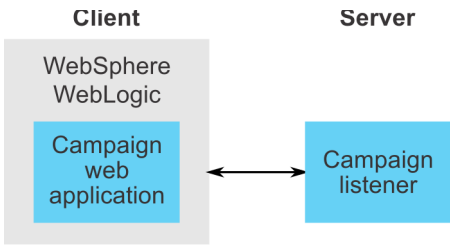
Example 1 - Communication between a browser and an IBM Marketing Software web application

When users communicate with IBM Marketing Software web applications through a browser, the browser is the client and the IBM Marketing Software web application is the server.



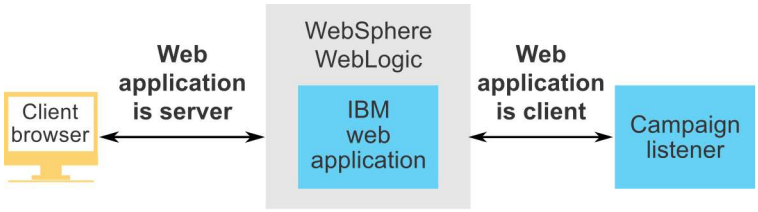
Example 2 - Communication between components of one IBM Marketing Software application

The two components of a single IBM Marketing Software application can also communicate with each other programmatically. For example, when the Campaign web application sends a request to the Campaign listener, the Campaign web application is the client and the listener is the server.



Example 3 - IBM Marketing Software components playing both roles

An IBM Marketing Software application component can communicate as a client in some exchanges and as a server in others. An example of these relationships is shown in the following diagram.



SSL in IBM Marketing Software

Many IBM application components can act as both server and client during normal operations, and some IBM components are written in Java and some in C++. These facts determine the format of the certificates you use. You specify the format when you create a self-signed certificate or purchase one from a CA.

IBM applications do not require a truststore when they act as a client making one-way SSL requests to an IBM server component.

Java component acting as a server

For IBM applications written in Java, using the JSSE SSL implementation, and deployed on an application server, you must configure the application server to use your certificate. The certificate must be stored in JKS format.

You cannot use the default certificate provided with the application server.

You can create JKS certificates for your Java applications using Java keytool.

C++ component acting as a server

The Campaign listener and Contact Optimization server component are written in C++, and require a .kdb certificate created in GSKit.

Java component acting as a client

For IBM applications written in Java and deployed on an application server, no truststore is needed. For ease of configuration, IBM Java applications acting as a client do not authenticate the server during one-way SSL communications. However, encryption does take place.

C/C++ components acting as a client

For applications written in C/C++ and using the GSkit implementation, no truststore is needed. The Campaign listener, Contact Optimization server component, and NetInsight fall into this category.

How many certificates?

Ideally, you should use a different certificate for every machine that hosts an IBM component acting as a server.

If you do not want to use multiple certificates, you can use the same certificate for all the IBM components acting as servers. If you use one certificate for all applications, when users access IBM applications for the first time, the browser asks whether they want to accept the certificate.

Related tasks:

"Configuring your web application servers for SSL" on page 162

Configuration process roadmap: implementing SSL in IBM Marketing Software

Use this configuration process roadmap to scan the tasks required to implement SSL in IBM Marketing Software. The Topic column provides links to the topics that describe the tasks in detail.

Table 51. Configuration process roadmap: implementing SSL in IBM Marketing Software

Topic	Information
"Creating and configuring certificates" on page 157	Obtain or create certificates.
"Configuring your web application servers for SSL" on page 162	Enable an SSL port in every application server where an IBM application is deployed. If you are not using the application server default certificate, configure it to use your certificate.
"Configuring IBM Marketing Software for SSL" on page 163	Set configuration properties in IBM Marketing Software.

Table 51. Configuration process roadmap: implementing SSL in IBM Marketing Software (continued)

Topic	Information
"Verifying your SSL configuration" on page 171	Log in to each of your IBM Marketing Software applications.

Certificates for SSL

This procedure describes how to create and configure your own certificates. Perform a similar procedure for every IBM Marketing Software you configure to use SSL. If you are configuring the Campaign + Engage integration, see the *IBM Campaign and Engage Integration Guide for IBM Marketing Cloud*.

You can obtain or create certificates in several ways. You can create self-signed certificates or you can obtain certificates from a certificate authority (CA).

Self-signed certificates

You can create self-signed certificates.

For C++ components acting as a server, use GSKit to create a `.kdb` certificate.

The Campaign listener implements SSL using the IBM GSKit library. The GSKit is installed withCampaign and it includes a command-line program called `gsk8capicmd_64` that can create a certificate file. For Java components acting as a server, use Java keytool to create a JKS certificate.

Certificates from a certificate authority

You can obtain certificates from a certificate authority (CA).

You can use the GS Kit to create requests that you can then send to a CA to create signed certificates. Or, you can obtain signed certificates entirely provided by the CA.

Consult your certificate authority documentation for instructions on how to obtain a signed certificate.

Creating and configuring certificates

This procedure describes how to create and configure your own certificates for your setup.

The following procedure describes how to create and configure self-signed certificates for Campaign and Marketing Platform.

For additional details on using GSKit, see http://www.ibm.com/developerworks/security/tutorials/se-gskit/. You can use these steps as a guide for configuring certificates for other IBM Marketing Software products.

1. Use GSKit to generate SSL certificates as follows.
   a. Create and initialize a new key database.

      For example:

`gsk8capicmd_64 -keydb -create -populate -db Campaign.kdb -pw password -stash` The `-stash` option is required for Campaign.

b. Use GSKit to generate a self-signed certificate for Campaign and store it in the key database, as follows.

   For example: `gsk8capicmd_64 -cert -create -db Campaign.kdb -dn "CN=example.com,O=domain,C=IN" -expire 3650 -pw password -size 1024 -label exampleLabel -default_cert yes`

c. Extract the public part of the certificate to a file.

   For the clients to trust a certificate, its public part needs to be distributed to the clients and stored in their key databases. In this step, you export the public part of the Campaign certificate. You import it in a later step.

   For example: `gsk8capicmd_64 -cert -extract -db Campaign.kdb -stashed -label exampleLabel -target Campaigncert.arm`

2. Import the `Campaigncert.arm` file into the application server where the Campaign web application is deployed.

   a. Copy the `Campaigncert.arm` file to the Campaign web application server.

   b. Add the `Campaigncert.arm` file in the **NodeDefaultTrustStore** of the WebSphere Application Server by completing the following steps:

      1) Click **Security** > **SSL Certificate and key management** > **Key stores and certificates**.

      2) Click **NodeDefaultTrustStore** > **Signer certificates**.

      3) Click **Add** and provide the **Alias** and the path where the `Key.arm` file is copied.

      4) Click **OK**.

3. Use keytool to generate and import a Java identity certificate for Marketing Platform as follows.

   a. Generate the Marketing Platform identity certificate.

      For example: `./keytool -genkey -alias PlatformClientIdentity -keyalg RSA -keystore PlatformClientIdentity.jks -keypass password -validity 3650 -dname "CN=example.com" -storepass password`

   b. Export the certificate from keystore (PlatformClientIdentity in the example) into a file (PlatformCertificate.cer in the example).

      `./keytool -export -keystore PlatformClientIdentity.jks -storepass password -alias PlatformClientIdentity -file PlatformCertificate.cer`

   c. Import the certificate that you exported into the Marketing Platform keystore (PlatformTrust.jks in the example).

      For example: `./keytool -import -alias PlatformClientIdentity -file PlatformCertificate.cer -keystore PlatformTrust.jks -storepass password`

4. Extract Personal and Signer certificates from the IBM Campaign WebSphere Application Server.

   a. Click **Security** > **SSL Certificate and key management** > **Key stores and certificates**.

   b. Click **NodeDefaultTrustStore** > **Personal certificates**.

   c. Select the default certificate.

   d. Provide the **Certificate file name** and a valid path in Campaign web application server, and click **OK**.

   e. Click **NodeDefaultTrustStore** > **Signer certificates**.

   f. Select the default certificate.

g. Provide the **Certificate file name** and a valid path in Campaign web application server, and click **OK**.

5. Import the Personal and Signer certificates into the listener keystore database.
   a. Copy the `ClientPersonal.cer` and `ClientSigner.cer` certificates to the Listener server.
   b. Import the Personal and Signer certificates to the listener keystore database by using the **gsk8capicmd_64** command from the location from where listener keystore database (`Campaigncert.kdb`) was created.

      ```
      gsk8capicmd_64 -cert -add -db Campaigncert.kdb -stashed -label ClientPersonalKey -file Clie
      gsk8capicmd_64 -cert -add -db Key.kdb -stashed -label ClientSignerlKey -file ClientSigner.c
      ```

6. Exchange certificates between Campaign and Marketing Platform.
   a. Use keytool to import the Campaign certificate into the Marketing Platform identity and keystore.

      For example: `./keytool -import -file` *`PATH_TO_GSKit_certificate`*`/Campaigncert.arm -alias listenerkey2 -keystore` *`PATH_TO_JAVA_CERTIFICATE`*`/PlatformClientIdentity.jks ./keytool -import -file` *`PATH_TO_GSKit_certificate`*`/Campaigncert.arm -alias listenerkey2 -keystore` *`PATH_TO_JAVA_CERTIFICATE`*`/PlatformTrust.jks`
   b. Use GSKit to import the Marketing Platform certificate into the Campaign key database.

      For example: `gsk8capicmd_64 -cert -add -db Campaign.kdb -stashed -label PlatformClientIdentity -file` *`PATH_TO_JAVA_CERTIFICATE`*`/PlatformCertificate.cer`

7. Use keytool to import the Java certificate and Campaign GSKit certificate into the application server Java keystore.
   a. Import the Marketing Platform Java certificate.

      For example: `./keytool -import -file` *`PATH_TO_JAVA_CERTIFICATE`*`/PlatformCertificate.cer -alias platformkey1 -keystore` *`APPLICATION_SERVER_HOME`*`/java/jre/lib/security/cacerts <Password: changeit>`
   b. Import the Campaign GSKit certificate.

      For example: `keytool -import -file` *`PATH_TO_GSKit_certificate`*`/campaign.arm -alias CampaignListener -keystore` *`APPLICATION_SERVER_HOME`*`/java/jre/lib/security/cacerts`

8. If the Campaign Listener is using SSL, In the **unicaACListener** settings set **useSSL** to **TRUE**, otherwise set it to **FALSE**.
9. Restart the Campaign Application Server and the Campaign Listener.

Creating and configuring certificates for a clustered environment

This procedure describes how to create and configure your own certificates for a clustered environment.

The Campaign web application must be configured for SSL by using default certificates.

The following procedure describes how to create and configure self-signed certificates for Campaign and Marketing Platform.

In a clustered environment where there is an IBM HTTP Server in front of the Campaign web application and IBM Campaign listener, follow these steps to configure the IBM Campaign listener in SSL.

For additional details on using GSKit, see http://www.ibm.com/developerworks/security/tutorials/se-gskit/. You can use these steps as a guide for configuring certificates for other IBM Marketing Software products.

This procedure is applicable for the default certificates that are provided by the IBM WebSphere Application Server. If you are using custom security certificates, you must follow the steps for the custom certificates used by the IBM WebSphere Application Server.

To configure the IBM HTTP Server in SSL, complete the following steps.

1. Use GSKit to generate SSL certificates as follows.
   a. Create and initialize a new key database.

      For example:

      `gsk8capicmd_64 -keydb -create -populate -db IHS.kdb -pw password -stash` The `-stash` option is required for Campaign.

   b. Use GSKit to generate a self-signed certificate for Campaign and store it in the key database, as follows.

      For example: `gsk8capicmd_64 -cert -create -db IHS.kdb -dn "CN=*.in.ibm.com" -expire 3650 -pw password -size 1024 -label key -default_cert yes`

   c. Extract the public part of the certificate to a file.

      For the clients to trust a certificate, its public part needs to be distributed to the clients and stored in their key databases. In this step, you export the public part of the Campaign certificate. You import it in a later step.

      For example: `gsk8capicmd_64 -cert -extract -db IHS.kdb -stashed -label key -target IHS.arm`

   d. Enable the following module in the `httpd.conf` file.

      For example:

      ```
      LoadModule ibm_ssl_module modules/mod_ibm_ssl.so

      Listen 443
      <VirtualHost *:443>
      SSLEnable
      </VirtualHost>
      KeyFile /data/webservers/IBM/IHS/ssl/IHS.kdb
      SSLStashFile /data/webservers/IBM/IHS/ssl/IHS.sth
      SSLDisable
      ```

   e. Provide the key file path in the `httpd.conf` file.
   f. Restart the IBM HTTP Server.
2. Generate keystore database files for the server that hosts the IBM Campaign listener.
   a. On the server that hosts the IBM Campaign listener, run the following commands from any location and note the path.

      ```
      gsk8capicmd_64 -keydb -create -populate -db Key.kdb -pw password -stash
      gsk8capicmd_64 -cert -create -db Key.kdb -dn "CN=*.in.ibm.com" -expire 3650 -pw password -siz
        -label key -default_cert yes
      gsk8capicmd_64 -cert -extract -db Key.kdb -stashed -label key -target Key.arm
      ```

   b. Verify that the following files are generated in the location from where you ran the above commands.

- Key.arm
- Key.crl
- Key.kdb
- Key.rdb
- Key.sth

3. Import the Key.arm and HIS.arm files into the application server where the Campaign web application is deployed.
   a. Copy the Key.arm and HIS.arm files to the Campaign web application server.
   b. Add the Key.arm and HIS.arm files in the **NodeDefaultTrustStore** of the WebSphere Application Server by completing the following steps:
      1) Click **Security** > **SSL Certificate and key management** > **Key stores and certificates**.
      2) Click **NodeDefaultTrustStore** > **Signer certificates**.
      3) Click **Add** and provide the **Alias** and the path where the Key.arm and HIS.arm files are copied.
      4) Click **OK**.
4. Extract the Personal and Signer certificates for the IBM WebSphere Application Server
   a. Click **Security** > **SSL Certificate and key management** > **Key stores and certificates**.
   b. Click **NodeDefaultTrustStore** > **Personal Certificates**.
   c. Select the default certificate.
   d. Add the Personal Certificate file name along with the valid path in the IBM Campaign web application server. For example, /opt/IBM/IBMMS101/ClientPersonal.cer.
   e. Click **OK**.
   f. Click **NodeDefaultTrustStore** > **Signer Certificates**.
   g. Select the default certificate.
   h. Add the Signer Certificate file name along with the valid path in the IBM Campaign web application server. For example, /opt/IBM/IBMMS101/ClientSigner.cer.
   i. Navigate to the folder and verify the both certificates are present in the folder.
5. Import the Personal and Signer certificates into the IBM Campaign listener and IBM HTTP Server keystore databases.
   a. Copy the ClientPersonal.cer and ClientSigner.cer certificates to the listener server. You can use the same location where the key.kdb file was created.
   b. Import the Personal and Signer certificates to the listener keystore database by using the **gsk8capicmd_64** command from the location where the listener keystore database (key.kdb) was created.

      ```
      gsk8capicmd_64 -cert -add -db Key.kdb -stashed -label ClientPersonalKey -file ClientPersona
      gsk8capicmd_64 -cert -add -db Key.kdb -stashed -label ClientSignerlKey -file ClientSigner.c
      ```

   c. Copy the ClientPersonal.cer and ClientSigner.cer certificates to the IBM HTTP Server. You can use the same location where the IHS.kdb file was created.
   d. Import the Personal and Signer certificates to the listener keystore database by using the **gsk8capicmd_64** command from the location where the IBM HTTP Server keystore database (IHS.kdb) was created.

6. Import the Campaign listener key in the IBM HTTP Server keystore database and import the IBM HTTP Server key in the Campaign keystore database.
   a. Copy the IBM HTTP Server key (`IHS.arm`) to the listener server.
   b. Import the IBM HTTP Server key to the listener keystore database by using the **gsk8capicmd_64** command from the location where the Campaign listener keystore database (`key.kdb`) was created.

      ```
      gsk8capicmd_64 -cert -add -db Key.kdb -stashed -label IHSKey -file IHS.arm
      ```

   c. Copy the Campaign listener key (`Key.arm`) to the listener server.
   d. Import the Campaign listener key to the IBM HTTP Server keystore database by using the **gsk8capicmd_64** command from the location where the IBM HTTP Server keystore database (`IHS.kdb`) was created.

      ```
      gsk8capicmd_64 -cert -add -db IHS.kdb -stashed -label IHSKey -file Key.arm
      ```

7. Restart the IBM Campaign application server, the IBM HTTP server and then start the IBM Campaign Listener.

Configuring your web application servers for SSL

On every application server on which an IBM Marketing Software application is deployed, configure the web application server to use the certificates you have decided to employ.

See your web application server documentation for details on performing these procedures.

Related concepts:

"SSL in IBM Marketing Software" on page 155

Ensuring cookie security

Some cookies may not be properly secured in the client browser. Not securing cookies leaves the application vulnerable to man-in-the-middle and session hijacking attacks. To fix this issue, take the following precautions.

- Enforce the use of SSL at all times to reduce the risk of cookies being intercepted on the wire.
- In the web application server, set the `secure` and `httponly` flags on all cookies.
  - The `secure` flag tells the browser to send the cookie only over an HTTPS connection. You must enable SSL on all applications that communicate with each other if you set this flag.
  - The `httponly` flag prevents cookies from being accessed through a client side script.

Setting the flags for SSL in WebSphere

To set the `secure` and `httponly` flags in WebSphere, use the following procedure.

You set the `secure` and `httponly` flags in the WebSphere administrative console.

Tip: See the WebSphere documentation for complete details.

1. At the application level for Marketing Platform, navigate to **Session Management** and click **Enable cookies**.
2. Check **Restrict cookies to HTTPS sessions** and **Set session cookies to HTTPOnly to help prevent cross-site scripting attacks**.
3. Save and apply your changes.

4. Stop and re-start the Marketing Platform application.

Setting the flags for SSL in WebLogic

To set the `secure` and `httponly` flags, use the following procedure.

Tip: See the WebLogic documentation for complete details.

1. If Marketing Platform is deployed and running, stop and undeploy it.
2. Extract the Marketing Platform WAR file.
3. Edit the `weblogic.xml` file to set the `secure` and `httponly` flags.
4. Recreate the Marketing Platform WAR file, redeploy, and re-start.

Configuring IBM Marketing Software for SSL

To configure IBM Marketing Software applications to use SSL, you must set some configuration properties. Use the procedures in this section that are appropriate for your installation of IBM Marketing Software products and the communications that you want to secure using SSL.

When you access your IBM Marketing Software installation over a secure connection, and when you set navigation properties for applications as described in the following procedures, you must use `https` and the secure port number in the URL. The default SSL port is `7002` for WebLogic and `8002` for WebSphere.

Configuring SSL in Marketing Platform

Follow this procedure to configure SSL in Marketing Platform.

1. Log in to IBM Marketing Software and click **Settings > Configuration**.
2. Set the value of the `General | Navigation | IBM Marketing Platform URL` property to Marketing Platform URL.

   For example: `https://`*`host.domain:SSL_port`*`/unica`

   where:

   - *`host`* is the name or IP address of the machine on which Marketing Platform is installed
   - *`domain`* is your company domain in which your IBM Marketing Software products are installed
   - *`SSL_Port`* is the SSL port in the application server on which Marketing Platform is deployed

   Note `https` in the URL.
3. Locate the properties under the `Navigation` category for each of your installed IBM Marketing Software products where you set the HTTP and HTTPS ports. The names of the properties might vary by product, but their purpose should be obvious. For each product, set these values to the HTTP and HTTPS port in the application server on which the product is deployed.
4. If you have implemented LDAP integration, perform the procedure described in "Configuring SSL in Marketing Platform with LDAP integration."
5. If you plan to use the data filtering feature, perform the procedure described in "Configuring SSL in Marketing Platform with data filters."

Related tasks:

"Configuring SSL in Marketing Platform with LDAP integration" on page 164

"Configuring SSL in Marketing Platform with data filters" on page 165

Configuring SSL in Platform for a clustered environment

Follow this procedure to configure SSL in IBM Platform in a clustered environment.

1. Log in to IBM Marketing Software and click **Settings** > **Configuration**.
2. Under `Affinium | Manager | Navigation`, set **IBM Marketing Platform** URL to the IBM Marketing Platform URL.

   For example: `https://<IHS_Host>/unica`.
3. Under `Affinium | Campaign | Navigation`, set **serverURL** to the IBM Campaign URL.

   For example: `https://<IHS_Host>/Campaign`.
4. Under `Affinium | Campaign | server`, set **fullContextPath** to the IBM Campaign URL.

   For example: `https://<IHS_Host>/Campaign`.
5. Under `Affinium | Campaign | unicaACListener`, set **serverhost** to the `<IHS Host>` and set **useSSL** to `True`.

Configuring SSL in Marketing Platform with LDAP integration

Follow this procedure to configure SSL in Marketing Platform.

1. Perform the procedure described in "Configuring SSL in Marketing Platform" if you have not done so already.
2. Log in to IBM Marketing Software and click **Settings > Configuration** .

   The Configuration page appears.
3. Navigate to the `IBM Marketing Software | IBM Marketing Platform | Security | Login Method details | LDAP` category and set the value of the `Require SSL for LDAP connection` property to `true`.

   This setting requires Marketing Platform to connect to the LDAP server using SSL when users log in.
4. Navigate to the `IBM Marketing Software | IBM Marketing Platform | Security | LDAP synchronization` category and set the following values.
   - Set the value of the `LDAP provider URL` property to: `ldaps://` *`host.domain:SSL_Port`*

     where:
     - *`host`* is the name or IP address of the LDAP server
     - *`domain`* is the domain of the LDAP server
     - *`SSL_Port`* is the SSL port of the LDAP server.

     For example: `ldaps://LDAPMachine.myCompany.com:636`

     Note the `ldaps` in the URL.

     The default SSL port for LDAP servers is `636`.
   - Set the value of the `Require SSL for LDAP connection` property to `true`.

     This setting requires Marketing Platform to connect to the LDAP server using SSL when it synchronizes with the LDAP server.

Related tasks:

"Configuring SSL in Marketing Platform" on page 163

Configuring SSL in Marketing Platform with data filters

When Marketing Platform is deployed with SSL and you plan to use the data filtering feature, you must perform this procedure to add the SSL options that perform hand shaking.

1. Perform the procedure described in "Configuring SSL in Marketing Platform" if you have not done so already.
2. Obtain the following.
   - A copy of the certificate file you created in Obtaining or creating certificates
   - The certificate password
3. Place the certificate file in the *JAVA_HOME*/jre/lib/security directory, where *JAVA_HOME* is the Java directory specified in the tools/bin/setenv script under your Marketing Platform installation.

   The setenv script specifies the Java instance used by Marketing Platform utilities.
4. Use the keytool program to import the certificate into the cacerts file for your Java instance.

   You can use the following example command as a guide.

   ```
   keytool -import -trustcacerts -file name_of_your_certificate.cer
   -keystore cacerts
   ```

   Enter the certificate password when prompted.

Related tasks:

"Configuring SSL in Marketing Platform" on page 163

Configuring SSL in Marketing Operations

Follow this procedure to configure SSL in Marketing Operations.

1. Log in to IBM Marketing Software and click **Settings > Configuration** .
2. Set the value of the Marketing Operations | navigation | serverURL property to the URL of the Marketing Operations web application.

   For example: serverURL=https://*host*:*SSL_port*/plan

   where:

   - *host* is the name or IP address of the machine on which Marketing Operations is installed.
   - *SSL_Port* is the SSL port of the Marketing Operations web application

   Note the https in the URL.
3. Open the plan_config.xml file in a text or XML editor.

   The plan_config.xml file is located in the conf directory under your Marketing Operations installation.
4. Set the UAPInitParam notifyPlanBaseURL property for your SSL connection.

   For example: <UAPInitParam notifyPlanBaseURL="https://*host*:*SSL_Port*/plan/affiniumplan.jsp"/>

   where:

   - *host* is the name or IP address of the machine on which Marketing Operations is installed.
   - *SSL_Port* is the SSL port of the Marketing Operations web application

   Note the https in the URL.

5. To enable Adobe Acrobat Online Markup functionality to work with Marketing Operations over HTTPS, set the markupServerURL property for your SSL connection.

   For example: <UAPInitParam markupServerURL="https://*host*:*SSLport*/plan/services/collabService?WSDL">

   where:

   - *host* is the name or IP address of the machine on which Marketing Operations is installed
   - *SSL_Port* is the SSL port of the Marketing Operations web application

   Note the https in the URL.
6. Save and close the plan_config.xml file.

Configuring SSL in Campaign

Follow this procedure to configure SSL in Campaign.

Note: If you are configuring SSL in Campaign, you must also configure the IBM Campaign Listener in SSL. If you do not set the IBM Campaign Listener in SSL the schedule flowchart status might be shown as Unknown.

1. Open the config.xml file in a text or XML editor.

   The config.xml file is in the conf directory under your Campaign installation.
2. Set the following values in the config.xml file.
   - Set unicaClientKeystore = *PATH_TO_JAVA_CERTIFICATE*/PlatformClientIdentity.jks
   - unicaClientKeystorePwd = *PASSWORD_OF_THE_JAVA_CERTIFICATE_FILE*
   - unicaServerSSLFile = *PATH_TO_GSKIT_CERTIFICATE*/campaign.kdb
   - unicaServerSSLFilePwd = *PATH_TO_GSKIT_STH_FILE*/*campaign.sth*
3. Save and close the config.xml file.
4. Log in to Marketing Platform and click **Settings > Configuration**.

   The Configuration page appears.
5. Set the value of the Campaign | unicaACListener | useSSL property to yes.
6. If you deployed the web application on an SSL port, set the value of the Campaign | navigation | serverURL property to the web application URL. For example:

   serverURL=https://host:SSL_port/Campaign

   where:

   - host is the name or IP address of the machine on which the web application is installed
   - SSL_Port is the SSL port of the web application

   Note the https in the URL.
7. If you are using the operational monitor, configure it for SSL by setting the value of the Campaign | monitoring | serverURL property to use HTTPS. For example:

   serverURL=https://host:SSL_port/Campaign/OperationMonitor

   where:

   - host is the name or IP address of the machine on which the web application is installed
   - SSL_Port is the SSL port of the web application

Note the https in the URL.

Configuring SSL in Campaign for a clustered environment

Follow this procedure to configure SSL in Campaign listener server in a clustered environment.

1. Open the config.xml file for the listener server in a text or XML editor.

   The config.xml file is in the conf directory under your Campaign installation.

2. Set the following values in the config.xml file.
   - Set **configurationServerBaseURL** to the Campaign SSL URL. This is the IBM HTTP Server URL.
   - Set **unicaServerSSLFile** to the path where the Key.kdb file is saved.
   - Set **unicaServerSSLFilePwd** to the path where the Key.sth file is saved.

   For example:

```
<configuration name="bootstrap">
   <category name="bootstrap">
      <property name="suiteName"><value>Affinium</value></property>
      <property name="clientType"><value>HTTP</value></property>
      <!-- configurationServerBaseURL value will be set by AffiniumSuite assembly installer --
      <property name="configurationServerBaseURL"><value>https://<IHS_Host>/Campaign</value></
      <property name="trustedApplication"><value>false</value></property>
      <property name="unicaClientKeystore"><value></value></property>
      <property name="unicaClientKeystorePwd"><value></value></property>
      <property name="unicaServerSSLFile"><value>/opt/IBM/IBMMS101/Campaign/bin/certs/Key.kdb<
      <property name="unicaServerSSLFilePwd"><value>/opt/IBM/IBMMS101/Campaign/bin/certs/Key.s
   </category>
</configuration>
```

3. Save and close the config.xml file.

Configuring IBM Campaign in SSL and IBM Campaign Listener in non-SSL

If your setup has IBM Campaign in SSL and the IBM Campaign Listener in the non-SSL mode, you must configure settings for the applications to work seamlessly.

The IBM Campaign web application must be configured in SSL by using the default certificates.

All configurations are applicable to the WebSphere Application Server for IBM Campaign. Multiple steps are involved to configure the SSL and non-SSL setup. Each step might have more substeps to be completed.

To configure IBM Campaign in SSL and IBM Campaign Listener in non-SSL, complete the following steps:

Complete the following steps.

Table 52. Configuring IBM Campaign in SSL and IBM Campaign Listener in non-SSL

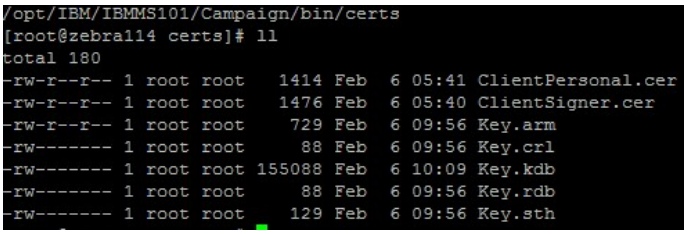
#	Step	Substeps

1	Generate keystore database files for the Listener server.	Run the following commands from and location and note the paths.
		`gsk8capicmd_64 -keydb -create -populate -db Key.kdb -pw password -stash`
		`gsk8capicmd_64 -cert -create -db Key.kdb -dn "CN=*.in.ibm.com" -expire 3650 -pw password -size 1024 -label key -default_cert yes`
		`gsk8capicmd_64 -cert -extract -db Key.kdb -stashed -label key -target Key.arm`
		
		The following files are generated at the location from where you ran the commands.
		`[root@zebra114 certs]# ll`
		`total 180`
		`-rw-r--r-- 1 root root 729 Feb 6 09:56 Key.arm`
		`-rw------- 1 root root 88 Feb 6 09:56 Key.crl`
		`-rw------- 1 root root 155088 Feb 6 10:09 Key.kdb`
		`-rw------- 1 root root 88 Feb 6 09:56 Key.rdb`
		`-rw------- 1 root root 129 Feb 6 09:56 Key.sth`

2	Import the `Key.arm` file into the application server where the Campaign web application is deployed.	1. Copy the `Key.arm` file to the Campaign web application server. 2. Add the `Key.arm` file in the **NodeDefaultTrustStore** of the WebSphere Application Server by completing the following steps: a. Click **Security** > **SSL Certificate and key management** > **Key stores and certificates**. b. Click **NodeDefaultTrustStore** > **Signer certificates**. c. Click **Add** and provide the **Alias** and the path where the `Key.arm` file is copied. d. Click **OK**. The listener key is added to the application server.
3.	Extract Personal and Signer certificates from the IBM Campaign WebSphere Application Server.	1. Click **Security** > **SSL Certificate and key management** > **Key stores and certificates**. 2. Click **NodeDefaultTrustStore** > **Personal certificates**. 3. Select the default certificate. 4. Provide the **Certificate file name** and a valid path in Campaign web application server, and click **OK**. 5. Click **NodeDefaultTrustStore** > **Signer certificates**. 6. Select the default certificate. 7. Provide the **Certificate file name** and a valid path in Campaign web application server, and click **OK**. The `ClientPersonal.cer` and `ClientSigner.cer` certificates are generated in the path that you provided.

Table 52. Configuring IBM Campaign in SSL and IBM Campaign Listener in non-SSL (continued)

#	Step	Substeps

4	Import the Personal and Signer certificates into the listener keystore database.	1. Copy the `ClientPersonal.cer` and `ClientSigner.cer` certificates to the Listener server.
		 
		2. Import the Personal and Signer certificates to the listener keystore database by using the **gsk8capicmd_64** command from the location from where listener keystore database (`Key.kdb`) was created.
		`gsk8capicmd_64 -cert -add -db Key.kdb -stashed -label ClientPersonalKey -file ClientPersonal.cer`
		`gsk8capicmd_64 -cert -add -db Key.kdb -stashed -label ClientSignerlKey -file ClientSigner.cer`

5	Modify `config.xml` file on the listener server.	Provide the following information:
		• **configurationServerBaseURL**: Provide the Campaign SSL URL.
		• **unicaServerSSLFile**: Provide the `.kdb` file path.
		• **unicaServerSSLFilePwd**: Provide the corresponding `.sth` file path.
		`<configuration name="bootstrap">`
		`<category name="bootstrap">`
		`<property name="suiteName"><value>Affinium</value></property>`
		`<property name="clientType"><value>HTTP</value></property>`
		`<!-- configurationServerBaseURL value will be set by AffiniumSuite ass`
		`<property name="configurationServerBaseURL">`
		`<value>https://eagle191.in.ibm.com:9447/Campaign</value>`
		`</property>`
		`<property name="trustedApplication"><value>false</value></property>`
		`<property name="unicaClientKeystore"><value></value></property>`
		`<property name="unicaClientKeystorePwd"><value></value></property>`
		`<property name="unicaServerSSLFile">`
		`<value>/opt/IBM/IBMMS101/Campaign/bin/certs/Key.kdb</value>`
		`</property>`
		`<property name="unicaServerSSLFilePwd">`
		`<value>/opt/IBM/IBMMS101/Campaign/bin/certs/Key.sth</value>`
		`</property>`
		`</category>`
		`</configuration>`

6	In the **unicaACListener** settings set **useSSL** to **TRUE**.	-
7	Restart the Campaign Application Server and the Campaign Listener.	-

Configuring SSL in Contact Optimization

Follow this procedure to configure SSL in Contact Optimization.

1. Open the `config.xml` file found in the `conf` directory of your Contact Optimization installation directory in a text or XML editor.
2. Set the value of `unicaServerSSLFile` to the full path of the certificate you are using.
3. Save and close the `config.xml` file.
4. Set the value of the `Campaign| unicaACOListener | useSSL` configuration property to `yes`.
5. If you are using the Contact Optimization command-line tool `ACOOptAdmin`, perform the following steps.
   a. Obtain the following.
      - A copy of the certificate file you created in Obtaining or creating certificates
      - The certificate password
   b. Place the certificate file in the *JAVA_HOME*`/jre/lib/security` directory, where *JAVA_HOME* is the Java directory specified in the `ACOOptAdmin` script.
   c. Use the `keytool` program to import the certificate into the `cacerts` file for your Java instance.

      You can use the following example command as a guide.

      `keytool -import -trustcacerts -file` *name_of_your_certificate.cer* `-keystore cacerts`

      Enter the certificate password when prompted.

Configuring SSL in Interact

You can configure SSL communication for Interact in three areas, although there is a significant performance cost if you do this.

The areas in that can use SSL are as follows.

- Design environment as the client and Runtime environment as the server.

  Use https in the URL referencing the Interact runtime server. For example, set `Campaign | partitions | partition[n] | Interact | ServerGroups | [serverGroup] | instanceURLs | [instanceURL] | instanceURL` to `https://myserver.domain.com:7007/interact`.
- Runtime environment as the client and Marketing Platform as the server.
- Your touchpoint as the client and the Runtime environment as the server.

  Specify the HTTPS URL with the `getInstance` method. If using a load balancer, you might need to configure your load balancer for SSL as well.
- if the Interact design server and Runtime server are on separate hosts using SSL, import the security certificates on the two servers to enable the SSL handshake to take place.

Important: There is a performance cost if you configure any part of Interact to communicate using SSL. IBM does not recommend configuring Interact to use SSL.

Configuring SSL in Distributed Marketing

After Campaign is configured to use SSL, no additional configuration is required to configure Distributed Marketing for SSL.

Configuring SSL in Reports

Follow this procedure to configure SSL in Reports.

1. Configure Cognos with SSL as described in the Cognos documentation.
2. Configure Apache with SSL as described in the Apache documentation.
3. Register the Cognos certificate with IBM Marketing Software as described in the Cognos documentation.
4. Register the IBM Marketing Software certificates with Cognos as described in the Cognos documentation.

Configuring SSL in Digital Analytics for On Premises

Digital Analytics for On Premises does not accept any requests: it always acts as the client in HTTP and HTTPS communications to resolve page titles on the web site being analyzed. If you need to resolve page titles for a site that uses SSL, you only need to ensure that the URL entered in the profile options for the website or clustered servers being analyzed is correct and that the URL includes the HTTPS protocol.

SDigital Analytics for On Premises does not communicate with Marketing Platform.

Verifying your SSL configuration

Follow this procedure to verify your SSL configuration.

1. Start each of your IBM Marketing Software applications.
2. Log in to IBM Marketing Software and access each of your installed IBM Marketing Software web applications.
3. For Interact runtime servers only, test the connection using the URL `https://host:port/interact/jsp/admin.jsp`.
4. If you are using a self-signed certificate, point your browser to each of the IBM Marketing Software server components and verify that the certificate information you receive is as expected.

   For example, if the Campaign listener is running on port 4664 on a host named `campaignHost`, point your browser to `https://campaignHost:4664`

   Your browser opens a window asking if you want to accept the certificate, and you can view certificate details.

Useful links for SSL

These links provide more information on the tasks required to implement SSL in IBM Marketing Software.

- GS Kit documentation - http://www.ibm.com/developerworks/security/tutorials/se-gskit/
- Java keytool documentation - http://docs.oracle.com/javase/7/docs/technotes/tools/solaris/keytool.html
- List of certificate authorities - http://www.dmoz.org/Computers/Security/Public_Key_Infrastructure/PKIX/Tools_and_Services/Third_Party_Certificate_Authorities/

Quality of protection (QoP) settings for WebLogic

You must set the QoP settings when you configure IBM Marketing Software applications to use SSL.

The following QoP settings are supported for WebLogic:

- TLS11
- TLS12

To change the QoP settings, complete the following steps:

Append the following option to the `JAVA_OPTIONS` variable:

- For TLS11- `-Dweblogic.security.SSL.protocolVersion=TLSv1.1`
- For TLS12- `-Dweblogic.security.SSL.protocolVersion=TLSv1.2`

Quality of protection (QoP) settings for WebSphere

You must set the QoP settings when you configure IBM Marketing Software applications to use SSL.

The following QoP settings are supported for WebSphere:

- SSL_TLS
- SSL
- TLS
- TLSv1
- SSL_TLSv2
- TLSv1.1
- TLSv1.2

To change the QoP settings, complete the following steps:

1. Go to **Security** > **SSL Certificate and key management** > **SSL configurations**
2. Select the required SSL configuration.
3. Under **Additional Properties**, click **Quality of protection (QoP) settings**.
4. In the Quality of protection (QoP) settings pane, select the required QoP settings from the drop-down list for **Protocol**.
5. Click **Save**.
6. In the `ssl.client.props` file located in the `WAS_install\profiles\AppSrv01\properties` folder, update the following:

   ```
   com.ibm.ssl.protocol=<Specify required QoP settings>
   ```
7. Restart the application server.

Chapter 16. Security framework for IBM Marketing Software APIs

Marketing Platform provides the security framework for the APIs implemented by IBM Marketing Software products.

A set of configuration properties on the **Settings > Configuration** page enables developers to set the following security for the APIs provided by IBM Marketing Software products.

- For a specific product API, you can block access to the product.
- For a specific product API, you can require HTTPS for communication between the specified API and the product.
- For a specific product API, you can require authentication for communication between the specified API and the product.

The configuration properties that control API security are located under the **IBM Marketing Platform | Security | API management** category. Each product has a configuration property template that you can use to create new security settings for the APIs provided by that product.

You can set and change the security settings for an API as appropriate for unit testing or deployment or during the overall lifecycle of APIs.

The security framework currently supports APIs for Campaign only.

The Marketing Platform security framework supports the following two authentication options for accessing protected APIs. You can use either one, depending on your environment.

- Internal users who are registered with Marketing Platform can be authenticated using their Marketing Platform login credentials to obtain a secure token.
- External users who are part of a federation that Marketing Platform is set up to use can be authenticated though the Identity Provider server.

Internal user authentication with the Marketing Platform login API

To authenticate internal users in client applications, use the Marketing Platform `login` API to generate secure tokens. You can then invoke any protected APIs by passing the required parameters in the request header, in addition to the parameters expected by the API itself.

The security filter intercepts these protected requests, validates them, and then passes them through for processing.

After the Marketing Platform user is authenticated, the Marketing Platform security filter adds the user's login name to the request as an attribute of the `USER_NAME_STRING` key before passing it to the product for processing.

© Copyright IBM Corp. 1999, 2018

The secure tokens have a default life span of 15 seconds. After the life span of the token expires, it cannot be used to invoke a protected API. Each time the Marketing Platform `login` API is invoked for a user, all previous security tokens for that user are invalidated.

You can change the life span of secure tokens by setting the value of the **Token lifetime** property located on the **Settings > Configuration** page under the **General | Miscellaneous** category.

Example URL

`http[s]://host:port/unica/api/manager/authentication/login/`

Header parameters

Table 53. Header parameters for the login API with internal users

Parameter	Description
`m_user_name`	The internal user's Marketing Platform login name.
`m_user_password`	The internal user's Marketing Platform password in plain text.

Response

When login succeeds, the response is HTTP 200 with the following JSON data.

- `m_tokenId` - randomly generated token
- `m_user_name` - user name of the logged in user
- `createDate` - timestamp in the format that is shown in the following example, where the time zone is IST:

  ```
  Mon Jul 06 18:23:35 IST 2015
  ```

When login fails with bad credentials, the response is HTTP 401 (unauthorized). When the `login` API is configured to be blocked, the response is 403 (forbidden). When the `login` API is configured to use HTTPS and if it is invoked on HTTP, the response is 403 (forbidden).

To log out internal users, use the Marketing Platform `logout` API.

Internal user logout with the Marketing Platform logout API

Use the Marketing Platform `logout` API to log out internal users and delete the secure token.

The `logout` API is protected by default. The authentication parameters are expected in the request header against pre-defined keys.

Example URL

`http[s]://host:port/unica/api/manager/authentication/logout/`

Header parameters

Table 54. Header parameters for the logout API

Parameter	Description
`m_user_name`	The internal user's Marketing Platform login name.
`m_tokenId`	The secure token obtained through authentication.
`api_auth_mode`	Use the value `manager` for internal users.

Response

When authentication succeeds, the response is `HTTP 200`, and the secure token is deleted. If the response is HTTP 200, the client application should confirm the logout.

When authentication fails, the response is `HTTP 401`.

External user authentication and logout through a federation

When Marketing Platform is integrated with a supported federation, users can log in to their own system, and the client application gets a token through the Identity Provider (IdP) server provided by Marketing Platform.

After a federated user is authenticated, their corresponding Marketing Platform login name is added to the request as an attribute of the `USER_NAME_STRING` key.

Log out should be done at the IdP server.

Header parameters

The following table describes the header parameters to use when authenticating through the IdP server provided by Marketing Platform.

Table 55. Header parameters with a federation

Parameter	Description
`f_userId`	User ID in the federation.
`f_clientId`	Client ID in the federation.
`f_spId`	Service provider ID in the federation.
`f_tokenId`	Single sign-on token from the IdP server.
`api_auth_mode`	Use the value `fsso` for federated authentication.

Response

The response is `HTTP 200`, with additional items depending on the API.

Related concepts:

Chapter 7, “SAML 2.0 based federated authentication,” on page 87

Related reference:

“IBM Marketing Platform | Security | API management | [Product] | (API configuration template)” on page 271

Chapter 17. Data filter creation and management

Data filters make it possible to restrict the customer data that an IBM Marketing Software user can view and work with in IBM Marketing Software applications. You can think of the data you secure with a data filter as a data set defined by the fields in your customer tables that you specify.

The various IBM Marketing Software applications use data filters in different ways. See the documentation for the individual products to determine whether the product uses data filtering, and if so, the details of how data filtering works within that product.

Overview of data filter creation

Marketing Platform provides the following features that IBM Marketing Software administrators use to set up data filters.

- A utility for defining data filters.
- A user interface for assigning users and groups to data filters and for viewing assigned data filters.

Data filter associations to restrict user access

To restrict data access for individual users or groups of users, you assign them to data filters. All IBM Marketing Software users and groups are available for assignment to data filters.

You can assign multiple users and groups to a single data filter, and you can also assign a user or a group of users to multiple data filters.

Note: Groups do not acquire the data filter assignments of their subgroups.

A user who is assigned to multiple data filters sees all of the records allowed by all of the data filters.

Two ways to create data filters: automatic generation and manual specification

Marketing Platform provides a utility, `datafilteringScriptTool`, that processes XML to create the data filters in the Marketing Platform system tables. Depending on how you write the XML, you can use this utility in two ways: automatic generation and manual specification.

Automatic generation

The `datafilteringScriptTool` utility can automatically generate data filters from a database table or view accessible using JDBC. The utility automatically creates data filters based on unique combinations of values in fields that you specify in the XML (one data filter for each unique combination).

You might want to use this method if you must create many data filters based on unique combinations of values in different fields.

© Copyright IBM Corp. 1999, 2018

Manual specification

The `datafilteringScriptTool` utility can create data filters one by one, based on field values that you specify.

You might want to use this method if you want to create a set of data filters that does not include every unique combination of field values.

Related concepts:

"Example: Manually specifying data filters" on page 189

"Example: Automatically generating a set of data filters" on page 193

Two ways to assign users and groups: in the user interface and in the XML

You have two options for assigning users and groups to data filters: through the user interface or in the XML you use to create the data filters. Assigning users in the XML is a useful method when you have many users, each of whom requires a separate filter.

Assigning users in the XML is available only when you create data filters using **manual specification**. When you assign users in the XML, you need the data filter IDs to specify the assignment, and these IDs are available only when you specify data filters using manual specification, not with automatic specification.

Details about using both methods for assigning users and groups are provided in this chapter.

Related concepts:

"About assigning users and groups in the XML" on page 198

"About assigning user and groups though the user interface" on page 203

Data filter concepts

To understand how to set up data filters, you need to be familiar with some concepts used in the data filter feature, in databases in general, and in Campaign in particular (if you are setting up data filters that will be used in an application in the Campaign family).

- **data configuration** - A data configuration groups a set of data filters. All data filters that secure related data are associated with the same data configuration.
- **audience** - The field or fields in customer tables designated in Campaign as an audience level. Typical audience levels are household and individual.
- **physical field name** - The physical names of fields in a database table are the names you see when you view the tables directly in the database client. When the data filter is in use, it uses the physical name when querying the customer database.
- **logical field name** - When you define data filters, you assign logical names to physical fields. If you are setting up data filters that will be used in an application in the Campaign family, these logical names must be the same as names assigned to fields in Campaign. This name is used by the utility when it generates data filters.

Configuration process roadmap: creating data filters

Use this configuration process roadmap to scan the tasks required to configure data filters. The Topic column provides links to the topics that describe the tasks in detail.

Table 56. Data filter configuration process roadmap

Topic	Information
• "Planning your data filter criteria: automatic generation" • "Planning your data filter criteria: manual generation" on page 180	Decide what customer data you want to secure.
"Obtaining the JDBC driver for your database: automatic generation only" on page 180	For automatic generation only: obtain the Type 4 JDBC driver that provides connectivity to the database containing the table on which you want to base your data filters.
"Obtaining required information" on page 181	Gather the required database information, and, if you plan to use the data filters with an application in the Campaign family, the Campaign-related information.
"Creating the XML to specify data filters" on page 181	Create the XML file that specifies the customer data used as criteria in each data filter.
"Setting required data filter configuration properties" on page 182	Set configuration properties that enable data filtering.
"Populating the data filter system tables" on page 183	Run the `datafilteringScriptToool` utility, which uses your XML to populate the Marketing Platform system tables that are used for data filters.
"Assigning users and groups to data filters" on page 183	If you do not assign users and groups to data filters within the XML, use the IBM Marketing Software data filter user interface to perform searches for users, groups, and data filters and then select items from the search results and assign them.

Planning your data filter criteria: automatic generation

Data filter criteria are based on your customer data. Before you can define data filters, you need to decide what customer data you want to secure.

For example, you might want to restrict access to customer data based on the countries, cities, and states where your customers live. If your customer database has a table that contains country, city, and state fields, you might choose to base a group of data filters on these fields. You would then use these values when you specify your data filters.

You should be aware of the following concepts when you plan how to create data filters using automatic generation.

- **profile field** - A field whose value is considered when the data filter generation utility looks for unique combinations of values. The utility creates a data filter for each unique combination of values. When the data filter is in effect in an IBM Marketing Software application, this value is used in a WHERE clause when customer records are queried. Because the clause tests for equality, profile fields must be defined against fields that support a finite set of distinct values.
- **fixed field** - An optional field that limits the records that the data filter generation utility looks at when querying for unique combinations of profile field values. The value you specify is also included in every generated data filter. When the data filter is in effect in an IBM Marketing Software application,

this value is used in a WHERE clause when customer records are queried. Because the clause tests for equality, fixed fields must be defined against fields that support a finite set of distinct values.

In the example above, you would probably create a fixed field for a country, and profile fields for city and state. The data filter generation utility creates a data filter for each unique combination of values it finds in these fields.

A IBM Marketing Software user assigned to one or more data filters would be able to view and work with only the data belonging to the customers who live in the countries, cities, and states represented by the assigned data filter(s).

It is possible that your customer tables do not contain every value for which you want to create a data filter. For example, you might not have customers in every country and state, but might want to prepare data filters for every country and state for future use. In that case, you can reference a table that includes every country and state and use it in the GenerateDataFilters section of your XML specification. When you have finished using the utility to create your data filters, you can discard this 'dummy' table.

Planning your data filter criteria: manual generation

Data filter criteria are based on your customer data. Before you can define data filters, you need to decide what customer data you want to secure.

For example, you might want to restrict access to customer data based on the geographical sales territory to which the IBM Marketing Software user is assigned. If the Region field in your customer database relates to your sales territories, you might choose to base a group of data filters on this field.

You should be aware of the concept of **field constraints**, which you need to understand when you plan how to create data filters using manual specification. A field constraint is a field/value pair used to specify a data filter. This value is used in a WHERE clause when customer records are queried. Because the clause tests for equality, field constraints must be defined against fields that support a finite set of distinct values.

In the example, the Region field might contain the following values: Asia, Europe, Middle East, North America, and South America. You use these values when you specify field constraints for your data filters. You would set up a different data filter for each of your sales territories, using the values in the Region field in your customer tables as field constraints.

A IBM Marketing Software user assigned to one or more data filters would be able to view and work with only the data belonging to the customers who fall within the sales territory or territories represented by the assigned data filter(s).

The data filters you create using the manual method can be assigned to users though the user interface of by making the assignments in the XML.

Obtaining the JDBC driver for your database: automatic generation only

A JDBC driver is required by the data filter generation utility (`datafilteringScriptTool`) when you use it to generate data filters automatically.

1. Obtain the Type 4 JDBC driver that provides connectivity to the database containing the table on which you want to base your data filters.
2. Place the driver on the machine where Marketing Platform is installed.
3. Make a note of the class name and path.

Obtaining required information

To create data filters, you need to gather information about your data and the way it is mapped in your IBM Marketing Software products.

For **manual specification** only: Obtain the following information.

- The physical name of the table containing the fields you want to use.
- The finite set of data in the fields you want to use for field constraints.
- If you plan to use the data filters in an application that is a member of the Campaign family, obtain the names assigned in Campaign to the following fields.
  - The audience fields
  - The fields you plan to use for field constraints.

For **automatic generation** only: Obtain the following information.

- For the database that contains the table you want to use in defining your data filters, the database type, the name or IP address, and the port.
- Database credentials (user name and password) that allow you to connect to the database.
- The physical name of the table containing the fields you want to use.
- The physical names of the fields you want to use for profile fields and fixed fields (fixed fields are optional).
- If you plan to use the data filters in an application that is a member of the Campaign family, obtain the names assigned in Campaign to the following fields.
  - The audience fields.
  - The fields you plan to use for fixed and profile fields.

Note: If you are defining data filters that will be used in an application that is a member of the Campaign family of products, the logical names of fields you specify in the XML that defines the data filters must match the names given to these fields in Campaign.

Creating the XML to specify data filters

Create the XML file that specifies the customer data used as criteria in each data filter. In the next step you will run a utility that populates the system tables with these specifications.

To create the data filters, the `datafilteringScriptTool` utility uses an XML representation of the data to insert entries into the Marketing Platform system table database.

Here is an overview of the elements in the XML that you create.

- `<Execute Batch>` - Command that initiates the data insertion process. This is repeated several times within the XML.
- `<AddDataConfiguration>` - Defines the data configurations, which are groups of related data filters.

- <AddLogicalFields> - Defines the fields on which to filter, and the data type of the fields.
- <AddDataFilter> - When you use **manual specification**, references a defined logical field, and specifies the field constraints.
- <GenerateDataFilters> - When you use **automatic specification**, references the fields and the values that limit the records considered for unique combinations of values used to define a set of data filters.
- <AddDataTable> - Defines the relationship between logical fields and their physical tables and columns. One logical field can apply to different physical tables, which allows one filter to apply to several tables.
- <addAudiences> - References a defined logical field, and specifies the audience level as defined in Campaign.
- <addAudienceTableAssociations> - Defines the relationship between an audience level and the defined table and the defined data filter configuration.
- <AddAssignments> - When you **create assignments within the XML rather than using the user interface**, associates individual users or groups of users with defined data filters.

For additional information, including descriptions of additional elements that are nested within the elements described above, see these topics in this chapter:

- The detailed descriptions of each element in the XML
- The XML provided in the example scenarios

Related concepts:

"Data filter XML reference" on page 183

"Example: Manually specifying data filters" on page 189

"Example: Automatically generating a set of data filters" on page 193

Setting required data filter configuration properties

Set required configuration properties to enable data filtering.

On the **Settings & Configuration** page, navigate to the **General | Data filtering** category and set the following properties.

- Default table name
- Default audience name

See each property's context help or the related topic link in this section for instructions on setting the values.

Related reference:

"IBM Marketing Software | General | Data filtering" on page 242

Optional configuration property to improve data filter performance

You can turn the data filter cache on for better performance.

To improve performance, set the value of the **General | Data filtering | Enable data filter cache** property to **true**. This property specifies whether Marketing Platform retrieves data filter definitions from the database or from a cache. When

this value is **true**, data filter definitions are stored in the cache and the cache is updated whenever there is any change in the data filter definitions.

You must restart the Marketing Platform web application after you make a change in this property value before it can take effect.

Related reference:

"IBM Marketing Software | General | Data filtering" on page 242

Populating the data filter system tables

Run the `datafilteringScriptTool` utility, which uses your XML to populate the data filter system tables.

For details on using the `datafilteringScriptTool` utility, see the full description elsewhere in this guide.

Note: If you need to delete data filters, run the `ManagerSchema_PurgeDataFiltering.sql` script, described elsewhere in this guide.

Related reference:

"datafilteringScriptTool" on page 226

"ManagerSchema_PurgeDataFiltering.sql" on page 236

Assigning users and groups to data filters

If you do not assign users or groups within the XML that you create, use the IBM Marketing Software data filter user interface to perform searches for users, groups, and data filters and then select items from the search results and assign them.

Related concepts:

"About assigning users and groups in the XML" on page 198

"About assigning user and groups though the user interface" on page 203

Data filter XML reference

This section describes the XML elements for which you must provide values.

About the IDs in the XML

Some objects require IDs. For example, data configurations, logical fields, and data tables all require that you specify IDs. The IDs you specify must be unique within a category of object.

Some objects reference other objects using IDs. For example, tables reference logical fields. When you need to reference another object, use the ID you specified for the object.

The XML uses the following convention for ID element names. This convention helps you understand when you must create a unique ID and when you must reference another ID within the XML.

- When you must create a unique ID, the element is named `id`.

- When you must reference another object ID, the element is named for the object. For example, the ID element where you reference a logical field is named `logicalFieldId`.

Note that the IDs you assign to an object are not the IDs Marketing Platform assigns to the object. The IDs you assign are used only for referencing the object within the XML.

AddDataConfiguration | dataConfiguration

This group of elements is used to define data configurations you use to group related data filters. You should create a data configuration for every set of related data filters.

Table 57. AddDataConfiguration | dataConfiguration

Element	Description	System table
`id`	Unique ID that you assign to this data configuration.	N/A
`name`	Name that you assign to this group of data filters.	Table: df_config Field: config_name

AddLogicalFields | logicalFields | LogicalField

This group of elements is used to define the logical fields corresponding to the fields in the customer table that you use to define your data filters. Create one logical field for each field from which you want to create field constraints, and one logical field for each audience.

Table 58. AddLogicalFields | logicalFields | LogicalField

Element	Description	System table
`id`	Unique ID that you assign to this logical field.	N/A
`name`	Logical name for this field or audience. If used with an application in the Campaign family, must be the same as the field or audience name used in Campaign.	Table: df_logical_field Field: logical_name
`type`	Data type of this field in the customer table. Allowed values are: • java.lang.String • java.lang.Long • java.lang.Double • java.lang.Boolean • java.lang.Date (The date format is month/day/year, where the month, day, and year are all expressed as numbers.)	Table: df_logical_field Field: type

GenerateDataFilters

This group of elements is used to generate data filters when you use **automatic generation**.

Table 59. GenerateDataFilters

Element	Description	System table
`tableName`	Physical name of the table from which you want to generate data filters, including the database schema name. If the database is case-sensitive, must match case used in the database.	Table: df_table Field: table_name
`configurationName`	Name of the data configuration in the `AddDataConfiguration | dataConfiguration` element with which this set of data filters is associated.	N/A
`jdbcUrl`	The URL reference for the customer database containing the table on which you want to base the data filters.	N/A
jdbcUser	The user name of an account with access to the customer database.	N/A
`jdbcPassword`	The password of the account with access to the customer database.	N/A
`jdbcDriverClass`	The name of the JDBC driver that provides connectivity to the customer database.	N/A
`jdbcDriverClassPath | string`	The path of the JDBC driver.	N/A

GenerateDataFilters | fixedFields | FixedField

This group of elements is used to specify the optional fields and the values that limit the records considered when the data filter generation utility looks for unique combinations of values to define a set of data filters. Used only when you use **automatic generation**.

Table 60. GenerateDataFilters | fixedFields | FixedField

Element	Description	System table
`expression`	One item of the data in the field that will be used in a WHERE clause when creating data filters and retrieving data for a user assigned to this filter. If the database is case-sensitive, must match case used in the database.	Table: df_field_constraint Field: expression
`logicalFieldName`	Name of the logical field in the AddLogicalFields | logicalFields | LogicalField element. This name appears as a label in the advanced search field in the Data Filter user interface in Marketing Platform.	Table: df_logical_field Field: logical_name

Table 60. GenerateDataFilters | fixedFields | FixedField (continued)

Element	Description	System table
`physicalFieldName`	Physical name of the field. If the database is case-sensitive, must match case used in the database.	N/A

GenerateDataFilters | profileField | ProfileField

This group of elements is used to specify fields whose unique combinations of values are used to define a set of data filters. Used only when you use **automatic generation**.

Table 61. GenerateDataFilters | profileField | ProfileField

Element	Description	System table
`logicalFieldName`	Name of the logical field in the `AddLogicalFields	logicalFields	LogicalField` element.	Table: df_logical_field Field: logical_name
`physicalFieldName`	Physical name of the field. If the database is case-sensitive, must match case used in the database.	N/A

AddDataTable | dataTable

This group of elements is used to assign IDs to customer tables.

Table 62. AddDataTable | dataTable

Element	Description	System table
`id`	Unique ID that you assign to this table.	N/A
`name`	Physical name of the customer table that you want to secure. If the database is case-sensitive, must match case used in the database.	Table: df_table Field: table_name

AddDataFilters | dataFilters | DataFilter

This group of elements is used to create a data filter when you use **manual specification**.

Table 63. AddDataFilters | dataFilters | DataFilter

Element	Description	System table
`configId`	ID of the data configuration in the AddDataConfiguration	dataConfiguration element with which this filter is associated.	N/A
`id`	Unique ID that you assign.	N/A

AddDataFilters | dataFilters | DataFilter | fieldConstraints | FieldConstraint

This group of elements is used to specify the data in a field used to define a data filter when you use **manual specification**.

Table 64. AddDataFilters | dataFilters | DataFilter | fieldConstraints | FieldConstraint

Element	Description	System table
`logicalFieldId`	ID of the logical field in the `AddLogicalFields | logicalFields | LogicalField` element.	N/A
`expression`	One item of the data in a field that is used in a `WHERE` clause when retrieving data for a user assigned to this filter. If the database is case-sensitive, must match case used in the database.	Table: df_fieldconstraint Field: expression

AddDataTable | dataTable | fields | TableField

This group of elements is used to map physical fields in the customer table to logical fields that you have defined.

Table 65. AddDataTable | dataTable | fields | TableField

Element	Description	System table
`name`	Physical name of the field in the customer table. If the database is case-sensitive, must match case used in the database.	Table: df_table_field Field: physical_name
`logicalFieldId`	ID of the logical field in the `AddLogicalFields | logicalFields | LogicalField` element.	N/A

AddAudience | audience

This group of elements is used to specify the name assigned in Campaign to an audience level used in the Campaign family of products.

Table 66. AddAudience | audience

Element	Description	System table
`id`	Unique ID that you assign to this audience.	N/A
`name`	Name of the audience as specified in Campaign.	Table: df_audience Field: audience_name

AddAudience | audience | fields | AudienceField

This group of elements is used to specify the field or fields in your customer tables that are used as audience fields.

Table 67. AddAudience | audience | fields | AudienceField

Element	Description	System table
`logicalFieldId`	ID of the logical field in the `AddLogicalFields	logicalFields	LogicalField` element. If used with an application in the Campaign family, must be the same logical name used in Campaign.	N/A
`fieldOrder`	For future use. Set the value to `0`.	N/A

addAudienceTableAssociations | addAudienceTableAssociation | audienceTableAssociation

This group of elements is used to associate pairs of audience fields and tables with data configurations. Create an association for every audience field.

Table 68. addAudienceTableAssociations | addAudienceTableAssociation | audienceTableAssociation

Element	Description	System table
`audienceId`	ID of the audience to be used in this association. Must be an ID value in an `AddAudience	audience` element.	N/A
`tableId`	ID of the table to be used in this association. Must be an ID value in an `AddDataTable	dataTable` element. The table must be one that contains the audience specified in the audienceID element. If the audience exists in more than one table, create multiple associations.	N/A
`configId`	ID of the data configuration to be used in this association. Must be an ID value in an `AddDataConfiguration	dataConfiguration` element.	N/A

AddAssignments | assignments | AssignmentByName

You can use this group of elements to associate users or groups with data filters. Optional. You can also make these assignments through the user interface.

Table 69. AddAssignments | assignments | AssignmentByName

Element	Description	System table
`namespaceId`	Name of the data configuration in the `AddDataConfiguration	dataConfiguration` element with which this set of data filters is associated.	N/A
`dataObjectId`	ID of the filter to be used in this association. Must be an ID value in a `DataFilter` element.	N/A

Table 69. AddAssignments | assignments | AssignmentByName (continued)

Element	Description	System table
`principalType`	The type of assignment. • 1 is for assigning a data filter to an individual user • 2 is for assigning a data filter to a group of users	Table: ols_assignment Field: principal_type
`principalName`	• If the value used for principalType is 1, set the value to the Marketing Platform login of the user you want to assign to the referenced data filter. • If the value used for principalType is 2, set the value to the name of the Marketing Platform group whose members you want to assign to the referenced data filter.	Table: ols_assignment Field: principal_id

Example: Manually specifying data filters

Jim needs to create a set of data filters based on sales territories.

In Campaign, the customer tables have already been mapped and audience levels have been defined.

Obtaining information

Jim determines that the Territory table contains the fields he needs to specify field constraints for the data filters.

The following table illustrates the information Jim obtains about the customer fields and their Campaign mappings.

Table 70. Territory table fields

Fields (physical name)	Fields (name in Campaign)	Data	Data type
cust_region	CustomerRegion	• Africa • Asia • Europe • Middle East • North America	java.lang.String
hh_id	HouseholdID	N/A	java.lang.Long
indiv_id	IndividualID	N/A	java.lang.Long

Jim learns that the audience names used in Campaign are household and individual. He notes that the Territory table contains two audience fields. The hh_id field corresponds to the household audience. The indiv_id field in the Territory table corresponds to the individual audience.

Because Jim must create one logical field for each audience, and one for the field constraint field, he knows he needs a total of three logical fields.

Jim also knows he needs to group the data filters in a data configuration. He decides to name his data configuration Territory.

Jim is now ready to create the XML.

Creating the XML

Here is the XML that Jim creates. Values based on the information he obtained are shown in **bold** .

```
<ExecuteBatch>
                <!-- ********************************** -->
                <!--          Data configuration        -->
                <!-- ********************************** -->
  <name>SeedData</name>
  <operations>
      <ExecuteBatch>
        <name>DataFilters</name>
        <operations>
          <AddDataConfiguration>
            <dataConfiguration>
                <id>1</id>
                <name>Territory</name>
            </dataConfiguration>
           </AddDataConfiguration>
         </operations>
       </ExecuteBatch>
                <!-- ********************************** -->
                <!--         Logical fields             -->
                <!-- ********************************** -->
     <AddLogicalFields>
       <logicalFields>
           <LogicalField>
               <id>1</id>
               <name>CustomerRegion</name>
               <type>java.lang.String</type>
           </LogicalField>
           <LogicalField>
               <id>2</id>
               <name>HouseholdID</name>
               <type>java.lang.Long</type>
           </LogicalField>
           <LogicalField>
               <id>3</id>
               <name>IndividualID</name>
               <type>java.lang.Long</type>
           </LogicalField>
           </logicalFields>
      </AddLogicalFields>
                <!-- ********************************** -->
                <!--    Territory field constraints     -->
                <!-- ********************************** -->
     <AddDataFilters>
       <dataFilters>
         <DataFilter>
           <configId>1</configId>
           <id>1</id>
           <fieldConstraints>
             <FieldConstraint>
               <logicalFieldId>1</logicalFieldId>
               <expression>Africa</expression>
             </FieldConstraint>
```

```
      </fieldConstraints>
    </DataFilter>
    <DataFilter>
      <configId>1</configId>
      <id>2</id>
      <fieldConstraints>
        <FieldConstraint>
          <logicalFieldId>1</logicalFieldId>
          <expression>Asia</expression>
        </FieldConstraint>
      </fieldConstraints>
    </DataFilter>
    <DataFilter>
      <configId>1</configId>
      <id>3</id>
      <fieldConstraints>
        <FieldConstraint>
          <logicalFieldId>1</logicalFieldId>
          <expression>Europe</expression>
        </FieldConstraint>
      </fieldConstraints>
    </DataFilter>
    <DataFilter>
      <configId>1</configId>
      <id>4</id>
      <fieldConstraints>
        <FieldConstraint>
          <logicalFieldId>1</logicalFieldId>
          <expression>Middle East</expression>
        </FieldConstraint>
      </fieldConstraints>
    </DataFilter>
    <DataFilter>
      <configId>1</configId>
      <id>5</id>
      <fieldConstraints>
        <FieldConstraint>
          <logicalFieldId>1</logicalFieldId>
          <expression>North America</expression>
        </FieldConstraint>
      </fieldConstraints>
    </DataFilter>
  </dataFilters>
</AddDataFilters>
          <!-- ********************************* -->
          <!--   Map physical to logical fields   -->
          <!-- ********************************* -->
<ExecuteBatch>
  <name>addTables</name>
  <operations>
    <AddDataTable>
      <dataTable>
      <id>1</id>
      <name>Territory</name>
      <fields>
        <TableField>
          <name>cust_region</name>
          <logicalFieldId>1</logicalFieldId>
        </TableField>
        <TableField>
          <name>hh_id</name>
          <logicalFieldId>2</logicalFieldId>
        </TableField>
        <TableField>
          <name>indiv_id</name>
          <logicalFieldId>3</logicalFieldId>
        </TableField>
```

```
            </fields>
            </dataTable>
          </AddDataTable>
        </operations>
      </ExecuteBatch>
            <!-- ******************************************************** -->
            <!--              Audience table associations                 -->
            <!-- ******************************************************** -->
       <ExecuteBatch>
             <name>addAudiences</name>
             <operations>
             <AddAudience>
               <audience>
               <id>1</id>
               <name>household</name>
               <fields>
                 <AudienceField>
                   <logicalFieldId>2</logicalFieldId>
                   <fieldOrder>0</fieldOrder>
                 </AudienceField>
               </fields>
             </audience>
           </AddAudience>
           <AddAudience>
               <audience>
               <id>2</id>
               <name>individual</name>
               <fields>
                 <AudienceField>
                   <logicalFieldId>3</logicalFieldId>
                   <fieldOrder>0</fieldOrder>
                 </AudienceField>
               </fields>
               </audience>
            </AddAudience>
            </operations>
       </ExecuteBatch>
            <!-- ******************************************************** -->
            <!--            Associate table-audience pairs                -->
            <!--               with data configuration                    -->
            <!-- ******************************************************** -->
       <ExecuteBatch>
             <name>addAudienceTableAssociations</name>
             <operations>
             <AddAudienceTableAssociation>
                 <audienceTableAssociation>
                   <audienceId>1</audienceId>
                   <tableId>1</tableId>
                   <configId>1</configId>
               </audienceTableAssociation>
             </AddAudienceTableAssociation>
             <AddAudienceTableAssociation>
                 <audienceTableAssociation>
                   <audienceId>2</audienceId>
                   <tableId>1</tableId>
                   <configId>1</configId>
                 </audienceTableAssociation>
             </AddAudienceTableAssociation>
             </operations>
       </ExecuteBatch>
    </operations>
</ExecuteBatch>
```

Populating the system tables

Jim has named his data filter XML file `regionDataFilters.xml` and saved it in the `tools/bin` directory under his Marketing Platform installation. He opens a command prompt and uses the datafilteringScriptTool utility to populate the data filter system tables.

Assigning users and groups to the data filters

Finally, Jim logs in to IBM Marketing Software with an account that has Admin access in Marketing Platform.

He knows that groups have already been set up in IBM Marketing Software with users assigned by region.

He goes to the Data Filter section and sees that the field constraints from his data filters are available in the advanced search for data filters. He performs a search for a data filter, using Africa as a search criterion. The data filter he set up for the Africa region appears in the search results.

Next, Jim performs a search for the Africa user group, which has been set up in IBM Marketing Software to hold all field marketers who are responsible for marketing to customers in Africa. The Africa group appears in the search results.

Jim then selects the group and the data filter in the search results, and assigns the group to the data filter by clicking the Assign button.

He continues to perform searches for data filters and groups until all assignments are completed.

Related concepts:

"About assigning user and groups though the user interface" on page 203

Example: Automatically generating a set of data filters

Jim needs to create a set of data filters based on countries, cities, and states.

In Campaign, the customer tables have already been mapped and audience levels have been defined.

Obtaining the JDBC driver

Jim knows that his company's customer database is Microsoft SQL server. He downloads the appropriate Type 4 driver and places it on the machine where the Marketing Platform is installed, making a note of the name and path of the driver.

- JDBC driver class name - `com.microsoft.sqlserver.jdbc.SQLServerDriver`
- JDBC driver path - `C:\tools\Java\MsJdbc\sqljdbc.jar`

Obtaining information

Jim obtains the name, host, and port of the customer database, and the credentials he needs to connect to it.

- Database name - Customers

- Database host name - companyHost
- Database port - 1433
- User name - sa
- Password - myPassword

Jim looks at the data in his company's customer database and sees that customers exist in every country, city, and state for which he wants to create a data filter. He determines that the Geographic table contains the fields he needs to specify fixed fields and profile fields for the data filters.

The following table illustrates the information Jim obtains about the customer fields and their Campaign mappings.

Table 71. Geographic table fields

Fields (Physical name)	**Fields (Name in Campaign)**	**Data**	**Data type**
country	Country	• USA • France • Britain	java.lang.String
city	City	A finite set of distinct cities	java.lang.String
state	State	A finite set of distinct states (or otherwise named regions, depending on country)	java.lang.String
hh_id	HouseholdID	N/A	java.lang.Long
indiv_id	IndividualID	N/A	java.lang.Long

Jim learns that the audience names used in Campaign are household and individual. He notes that the Geographic table contains two audience fields.

- The `hh_id field` corresponds to the household audience.
- The `indiv_id` field in the Geographic table corresponds to the individual audience.

Because Jim must create one logical field for each audience, and one for each of the fixed and profile fields, he knows he needs a total of five logical fields.

Jim also knows he needs to group the data filters in a data configuration. He decides to name his data configuration Geographic.

Jim is now ready to create the XML.

Creating the XML

Here is the XML that Jim creates. Values based on the information he obtained or decided to use are shown in **bold**.

```
<ExecuteBatch>
                <!-- ********************************** -->
                <!--          Data configuration        -->
                <!-- ********************************** -->
  <name>SeedData</name>
```

```
<operations>
    <ExecuteBatch>
      <name>DataFilters</name>
      <operations>
          <AddDataConfiguration>
              <dataConfiguration>
                  <id>1</id>
                  <name>Geographic</name>
              </dataConfiguration>
              </AddDataConfiguration>
      </operations>
    </ExecuteBatch>
              <!-- ********************************** -->
              <!--         Logical fields             -->
              <!-- ********************************** -->
    <AddLogicalFields>
      <logicalFields>
          <LogicalField>
              <id>1</id>
              <name>Country</name>
              <type>java.lang.String</type>
          </LogicalField>
          <LogicalField>
              <id>2</id>
              <name>City</name>
              <type>java.lang.String</type>
          </LogicalField>
          <LogicalField>
              <id>3</id>
              <name>State</name>
              <type>java.lang.String</type>
          </LogicalField>
          <LogicalField>
              <id>4</id>
              <name>HouseholdID</name>
              <type>java.lang.Long</type>
          </LogicalField>
          <LogicalField>
              <id>5</id>
              <name>IndividualID</name>
              <type>java.lang.Long</type>
          </LogicalField>
          </logicalFields>
    </AddLogicalFields>
              <!-- ********************************** -->
              <!--       Generate data filters        -->
              <!-- ********************************** -->
    <GenerateDataFilters>
              <!-- ******************************************************** -->
              <!-- Specify the table to be scanned for unique combinations -->
              <!-- of values  from which data filters will be defined.     -->
              <!-- ******************************************************** -->
            <tableName>Geographic</tableName>
              <!-- ******************************************************** -->
              <!-- Identify the data configuration  with which             -->
              <!-- generated data filters will be associated.              -->
              <!-- ******************************************************** -->
           <configurationName>Geographic</configurationName>
            <!-- Specify the data source connection information. -->
           <jdbcUrl>
             jdbc:sqlserver://localhost:1433;databaseName=Customers
           </jdbcUrl>
           <jdbcUser>sa</jdbcUser>
           <jdbcPassword>myPassword</jdbcPassword>
           <jdbcDriverClass>
                  com.microsoft.sqlserver.jdbc.SQLServerDriver</jdbcDriverClass>
           <jdbcDriverClassPath>
```

```
            <string>C:\tools\Java\MsJdbc\sqljdbc.jar</string>
        </jdbcDriverClassPath>
           <!-- ********************************** -->
           <!--      Specify the fixed fields      -->
           <!-- ********************************** -->
         <fixedFields>
             <FixedField>
                <expression>USA</expression>
                <logicalFieldName>Country</logicalFieldName>
                <physicalFieldName>country</physicalFieldName>
             </FixedField>
             <FixedField>
                <expression>France</expression>
                <logicalFieldName>Country</logicalFieldName>
                <physicalFieldName>country</physicalFieldName>
             </FixedField>
             <FixedField>
                <expression>Britain</expression>
                <logicalFieldName>Country</logicalFieldName>
                <physicalFieldName>country</physicalFieldName>
             </FixedField>
         </fixedFields>
         <!-- Specify the profile fields. -->
         <profileFields>
             <ProfileField>
                <logicalFieldName>State</logicalFieldName>
                <physicalFieldName>state</physicalFieldName>
             </ProfileField>
             <ProfileField>
                <logicalFieldName>City</logicalFieldName>
                <physicalFieldName>city</physicalFieldName>
             </ProfileField>
         </profileFields>
</GenerateDataFilters>
          <!-- ********************************** -->
          <!--   Map physical to logical fields   -->
          <!-- ********************************** -->
<ExecuteBatch>
        <name>addTables</name>
        <operations>
        <AddDataTable>
             <dataTable>
             <id>1</id>
             <name>Geographic</name>
             <fields>
                <TableField>
                       <name>country</name>
                       <logicalFieldId>1</logicalFieldId>
                </TableField>
                <TableField>
                       <name>city</name>
                       <logicalFieldId>2</logicalFieldId>
                </TableField>
                <TableField>
                       <name>state</name>
                       <logicalFieldId>3</logicalFieldId>
                </TableField>
                <TableField>
                       <name>hh_id</name>
                       <logicalFieldId>4</logicalFieldId>
                </TableField>
                <TableField>
                       <name>indiv_id</name>
                       <logicalFieldId>5</logicalFieldId>
                </TableField>
             </fields>
             </dataTable>
```

```
                </AddDataTable>
            </operations>
        </ExecuteBatch>
            <!-- ********************************************************* -->
            <!--              Audience table associations                  -->
            <!-- ********************************************************* -->
        <ExecuteBatch>
            <name>addAudiences</name>
            <operations>
            <AddAudience>
                <audience>
                <id>1</id>
                <name>household</name>
                <fields>
                   <AudienceField>
                      <logicalFieldId>4</logicalFieldId>
                      <fieldOrder>0</fieldOrder>
                   </AudienceField>
                </fields>
                </audience>
            </AddAudience>
            <AddAudience>
                <audience>
                <id>2</id>
                <name>individual</name>
                <fields>
                   <AudienceField>
                     <logicalFieldId>5</logicalFieldId>
                     <fieldOrder>0</fieldOrder>
                   </AudienceField>
                </fields>
                </audience>
            </AddAudience>
            </operations>
        </ExecuteBatch>
            <!-- ********************************************************* -->
            <!--            Associate table-audience pairs               -->
            <!--               with data configuration                   -->
            <!-- ********************************************************* -->
        <ExecuteBatch>
            <name>addAudienceTableAssociations</name>
            <operations>
            <AddAudienceTableAssociation>
               <audienceTableAssociation>
                   <audienceId>1</audienceId>
                   <tableId>1</tableId>
                   <configId>1</configId>
               </audienceTableAssociation>
            </AddAudienceTableAssociation>
            <AddAudienceTableAssociation>
               <audienceTableAssociation>
                   <audienceId>2</audienceId>
                   <tableId>1</tableId>
                   <configId>1</configId>
               </audienceTableAssociation>
            </AddAudienceTableAssociation>
            </operations>
        </ExecuteBatch>
    </operations>
</ExecuteBatch>
```

Populating the system tables

Jim has named his data filter XML file `geographicDataFilters.xml` and saved it in `tools/bin` directory under his Marketing Platform installation. He opens a command prompt and uses the `datafilteringScriptTool` utility to populate the data filter system tables.

The utility creates many data filters. In each data filter, the criteria are a country (the fixed field) and a unique combination of city and state obtained when the utility queried the database for records containing the fixed field value. All unique combinations of city and state are used for each country specified as a fixed field.

Assigning users and groups to the data filters

Finally, Jim logs in to the Marketing Platform with an account that has Admin access in Marketing Platform.

He knows that groups have already been set up in Marketing Platform with users assigned by city.

He goes to the Data Filter section and sees that the country, city, and state values from his data filters are available in the advanced search for data filters. He performs a search for a data filter, using Boston, a city in the USA, as a search criterion. The data filter for Boston appears in the search results.

Next, Jim performs a search for the Boston user group, which has been set up in Marketing Platform to hold all field marketers who are responsible for marketing to customers in Boston. The Boston group appears in the search results.

Jim then selects the group and the data filter in the search results, and assigns the group to the data filter by clicking the Assign button.

He continues to perform searches for data filters and groups until all assignments are completed.

Related concepts:

"About assigning user and groups though the user interface" on page 203

About assigning users and groups in the XML

You can assign users or groups to data filters in the XML, as an alternative to doing this through the user interface. Assigning users and groups to data filters in the XML is available only when you use manual specification to create the data filters.

You can use a wild card, `#user_login#`, that automatically creates data filters based on the user's Marketing Platform login name.

You use the `AddAssignments` XML element block to associate users or groups with your data filters.

Scenario used in the example

The example uses the following scenario.

An organization uses Distributed Marketing and wants to create data filters that allow field marketers to see only the customers in the region to which they are assigned. Thus, each user requires his or her own data filter.

In Distributed Marketing the list display and the form templates are set up based on region. This configuration is described in more detail in the *IBM Distributed Marketing Administrator's Guide*.

The audience level is Customer.

The data filters are created against four tables in the `exampleSchema` database, as described in the following table.

Table 72. Tables and fields used in the examples

Table	Fields
`exampleSchema.Corporate_Lists`	`UserID`, `State`, and `RegionID` This is the list display table set up in Distributed Marketing. The UserID column contains the Marketing Platform login names of the field marketers. This table associates field marketer Marketing Platform login names with their assigned region.
`exampleSchema.customer_contact`	`Indiv_ID`, `Region_ID`, and `State` fields for customers
`exampleSchema.lkup_state`	A lookup table for the `state_name` field
`exampleSchema.lkup_region`	A lookup table for the `region_id` field

Example: Using the wild card to assign group members to data filters

To create a separate data filter for each member of a specified group, you do the following.

- Create logical fields as usual.
- Create a single data filter with the wild card `#user_login#` in the `expression` element.
- Under the `AssignmentByName` element, set the `principalType` to `2`, set the `principalName` element to the group name, and set the `dataObjectId` element to the ID of the wild card data filter.
- Create audience associations as usual.

The following XML illustrates this method, using the scenario described above.

```
<ExecuteBatch>
               <!-- ********************************** -->
               <!--          Data configuration        -->
               <!-- ********************************** -->
    <name>SeedData</name>
    <operations>
        <ExecuteBatch>
            <name>DataFiltering</name>
            <operations>
                <AddDataConfiguration>
                    <dataConfiguration>
                        <id>1</id>
                        <name>collaborate</name>
                    </dataConfiguration>
```

```
        </AddDataConfiguration>
    </operations>
</ExecuteBatch>
        <!-- ********************************** -->
        <!--         Logical fields             -->
        <!-- ********************************** -->
<AddLogicalFields>
    <logicalFields>
        <LogicalField>
            <id>1</id>
            <name>Customer_ID</name>
            <type>java.lang.String</type>
        </LogicalField>

        <LogicalField>
            <id>2</id>
            <name>AudienceLevel</name>
            <type>java.lang.String</type>
        </LogicalField>

        <LogicalField>
            <id>3</id>
            <name>UserID</name>
            <type>java.lang.String</type>
        </LogicalField>

        <LogicalField>
            <id>4</id>
            <name>State_code</name>
            <type>java.lang.String</type>
        </LogicalField>

        <LogicalField>
            <id>5</id>
            <name>Region</name>
            <type>java.lang.Long</type>
        </LogicalField>
    </logicalFields>
</AddLogicalFields>
        <!-- ********************************** -->
        <!--       Wild card data filter        -->
        <!-- ********************************** -->
<AddDataFilters>
    <dataFilters>
        <DataFilter><
            <configId>1</configId>
            <id>1</id>
            <fieldConstraints>
                <FieldConstraint>
                    <logicalFieldId>3</logicalFieldId>
                    <expression>#user_login#</expression>
                </FieldConstraint>
            </fieldConstraints>
        </DataFilter>
    </dataFilters>
</AddDataFilters>
        <!-- ********************************************************** -->
        <!--                      Add data tables                       -->
        <!-- ********************************************************** -->

<ExecuteBatch>
    <name>addTables</name>
    <operations>
        <!-- ********************************************************** -->
        <!--           Table exampleSchema.Corporate_Lists              -->
        <!-- ********************************************************** -->
         <AddDataTable>
```

```
    <dataTable>
        <id>1</id>
        <name>exampleSchema.Corporate_Lists</name>
        <fields>
            <TableField>
                <tableId>1</tableId>
                <name>UserID</name>
                <logicalFieldId>3</logicalFieldId>
            </TableField>
            <TableField>
                <tableId>1</tableId>
                <name>State</name>
                <logicalFieldId>4</logicalFieldId>
            </TableField>
            <TableField>
                <tableId>1</tableId>
                <name>Region_ID</name>
                <logicalFieldId>5</logicalFieldId>
            </TableField>
        </fields>
    </dataTable>
</AddDataTable>
<!-- ********************************************************* -->
<!--           Table exampleSchema.customer_contact            -->
<!-- ********************************************************* -->
<AddDataTable>
    <dataTable>
        <id>2</id>
        <name>exampleSchema.customer_contact</name>
        <fields>
            <TableField>
                <tableId>2</tableId>
                <name>Indiv_ID</name>
                <logicalFieldId>1</logicalFieldId>
            </TableField>
            <TableField>
                <tableId>2</tableId>
                <name>Region_ID</name>
                <logicalFieldId>5</logicalFieldId>
            </TableField>
            <TableField>
                <tableId>2</tableId>
                <name>State</name>
                <logicalFieldId>4</logicalFieldId>
            </TableField>
        </fields>
    </dataTable>
</AddDataTable>
<!-- ********************************************************* -->
<!--              Table exampleSchema.lkup_state               -->
<!-- ********************************************************* -->
<AddDataTable>
    <dataTable>
        <id>3</id>
        <name>example.schema.lkup_state</name>
        <fields>
            <TableField>
                <tableId>3</tableId>
                <name>state_name</name>
                <logicalFieldId>4</logicalFieldId>
            </TableField>
        </fields>
    </dataTable>
</AddDataTable>
<!-- ********************************************************* -->
<!--             Table exampleSchema.lkup_region               -->
<!-- ********************************************************* -->
```

```
        <AddDataTable>
            <dataTable>
                <id>4</id>
                <name>exampleSchema.lkup_region</name>
                <fields>
                    <TableField>
                        <tableId>4</tableId>
                        <name>Region_ID</name>
                        <logicalFieldId>5</logicalFieldId>
                    </TableField>
                </fields>
            </dataTable>
        </AddDataTable>
    </operations>
</ExecuteBatch>
        <!-- ******************************************************** -->
        <!--              Audience table associations                 -->
        <!-- ******************************************************** -->
<ExecuteBatch>
    <name>addAudiences</name>
    <operations>
        <AddAudience>
            <audience>
                <id>1</id>
                <name>Customer</name>
                <fields>
                    <AudienceField>
                        <logicalFieldId>2</logicalFieldId>
                        <fieldOrder>0</fieldOrder>
                    </AudienceField>
                </fields>
            </audience>
        </AddAudience>

        <AddAudience>
            <audience>
                <id>2</id>
                <name>default</name>
                <fields>
                    <AudienceField>
                        <logicalFieldId>2</logicalFieldId>
                        <fieldOrder>0</fieldOrder>
                    </AudienceField>
                </fields>
            </audience>
        </AddAudience>
    </operations>
</ExecuteBatch>

<ExecuteBatch>
    <name>addAudienceTableAssociations</name>
    <operations>
        <AddAudienceTableAssociation>
            <audienceTableAssociation>
                <audienceId>1</audienceId>
                <tableId>1</tableId>
                <configId>1</configId>
            </audienceTableAssociation>
        </AddAudienceTableAssociation>

        <AddAudienceTableAssociation>
            <audienceTableAssociation>
                <audienceId>1</audienceId>
                <tableId>2</tableId>
                <configId>1</configId>
            </audienceTableAssociation>
        </AddAudienceTableAssociation>
```

```
                <AddAudienceTableAssociation>
                    <audienceTableAssociation>
                        <audienceId>2</audienceId>
                        <tableId>3</tableId>
                        <configId>1</configId>
                    </audienceTableAssociation>
                </AddAudienceTableAssociation>

                <AddAudienceTableAssociation>
                    <audienceTableAssociation>
                        <audienceId>2</audienceId>
                        <tableId>4</tableId>
                        <configId>1</configId>
                    </audienceTableAssociation>
                </AddAudienceTableAssociation>

            </operations>
        </ExecuteBatch>
                <!-- ******************************************************** -->
                <!--           Link filters (dataObjectId) to group           -->
                <!-- ******************************************************** -->
        <AddAssignments>
            <assignments>
                 <AssignmentByName>
                    <namespaceId>1</namespaceId>
                    <dataObjectId>1</dataObjectId>
                    <principalType>2</principalType>
                    <principalName>FieldMarketer</principalName>
                </AssignmentByName>
            </assignments>
        </AddAssignments>
    </operations>
</ExecuteBatch>
```

About assigning user and groups though the user interface

You can assign users and groups to data filters on the **Settings > Data filters** pages.

To work with data filters on the **Settings > Data filters** pages, the following must be true.

- The data filters must be set up in Marketing Platform system tables.
- You must log in as a user with the Marketing Platform permission **Administer data filters page**. The pre-configured Marketing Platform **AdminRole** role has this permission.

Advanced search

Marketing Platform provides a user interface for assigning users and groups to data filters. This user interface relies on an advanced search feature to obtain lists of users, groups, and data filters. You can select users and groups from these lists and assign them to data filters that you select.

Data filter search

The search feature for data filters provides search criteria that are the same as the criteria specified when the data filters were set up. For example, suppose a set of data filters is based on a field containing the following data relating to sales territories.

- Africa

- Asia
- Europe
- Middle East
- North America

The data filter advanced search would provide this data in a drop-down list from which you can select when searching for data filters.

User and group search

The advanced search feature for users and groups provides a text field where you can enter text for the search to match.

When a tab containing the user and group advanced search first loads, there is a wildcard (*) in both the User and Group text fields. A search performed using this wildcard returns all records.

If you delete the wildcard and do not enter any other text, leaving the field blank, no records are returned. For example, if you perform a search with the User text field blank and an asterisk in the Group text field, only groups would be listed in the results.

On the View Assignments tab, if you leave both the User and Group text fields blank, no records are returned regardless of what data filter criteria are selected.

When you enter text in the field, the search matches the characters you enter in the text field, in the order you enter them. For example, to obtain a group named North America, you could enter any letter or group of letters (in order) that occurs in the name. You would obtain North America in the results if you entered "north" or "h", but not if you entered "htron."

The search is not case-sensitive. That is, "North" is the same as "north."

Viewing assigned data filters

Use this procedure to view assigned data filters

1. Log in to IBM Marketing Software as a user with the Marketing Platform AdminRole role and click **Data Filtering**.

   The Data filters page displays.
2. Click **View assigned data filters**.
3. Perform an advanced search for assigned data filters to obtain search results.

A list of data filters that meet the criteria is displayed.

Assigning users and groups to data filters

Use this procedure to assign users and groups to data filters.

1. Log in to IBM Marketing Software as a user with the Marketing Platform AdminRole role and click **Settings > Data filters**.

   The Data filters page displays.
2. Click **Assign users or groups**.
3. Perform an advanced search for data filters to obtain a list of data filters.
4. Perform an advanced search for the users, groups, or both to obtain a list of users and groups.

5. From your search results lists, select data filters and the users and groups you want to assign to them.
6. Click **Assign**.

The selected users and groups are assigned to the selected data filters.

Removing data filter assignments

Use this procedure to remove data filter assignments.

1. Log in to IBM Marketing Software as a user with the Marketing Platform AdminRole role and click **Settings > Data filters**.

   The Data Filters page displays.
2. Click **View assigned data filters**.
3. Perform an advanced search for assigned data filters to obtain search results from which you want to select.
4. From your search results list, select the data filters whose assignments you want to delete.
5. Click **Unassign**.

The selected assignments are deleted. The data filters themselves are not deleted.

Adding data filters after the initial set has been created

You can continue to add data filters after you have created the initial set. For example, you might create a set of data filters based on countries and their city/state combinations, and later decide to create another set based on zip codes.

You can obtain the XML for additional data filters in either of the following ways.

- Modify your original XML file to add new filters. When you seed the database using the `dataFilteringScriptTool` utility, the Marketing Platform creates only the new data filters.
- Create an XML file specifying new data filters. When you seed the database using the `dataFilteringScriptTool` utility, existing data filters are not deleted.

After you have created the XML, populate the data filter tables and assign users and groups as described in this guide.

Chapter 18. Audit event tracking in IBM Marketing Software

You can configure which audit events are tracked and assign a severity level to each tracked event.

Two kinds of audit events are tracked:

- Security related events such as changes to user status, group memberships, and permissions
- Changes to IBM Marketing Software configuration properties that are managed on the **Settings > Configuration** page

The audit event information is independent of the system log, and configuration you perform for the system log does not affect audit event tracking.

The Audit Events report provides a convenient way to view the tracked events. You can configure the content of the report, filter the information shown in the report, and export report data.

You must have the AdminRole or PlatformAdminRole role in Marketing Platform to configure the Audit Events report and audit backups or to view the report.

Limitations on audit event tracking

If you configure tracking for configuration property audit events, these changes are tracked only when they are performed using the **Settings > Configuration** page.

For example, the following configuration property changes are not tracked.

- Changes made using Marketing Platform utility `configTool`
- Changes made during installation and upgrade of IBM Marketing Software products

Also, when you manually add default users, roles, and permissions for Marketing Platform and Campaign using the Marketing Platform `populateDB` utility, these changes are not tracked.

Legacy audit events

Previous releases of Marketing Platform saved audit events in the Marketing Platform system tables, although no report was available. If you upgrade from a version earlier than 9.1.1, the Audit Events report includes these legacy events.

Legacy audit events are displayed in the report as follows.

- The **Severity** column contains **No severity (Legacy)** to indicate that these audit records were stored before the audit report was available.
- In an environment with just one partition, the **Partition** column contains the ID of the default partition.
- In a multi-partition environment, the **Partition** column contains **-1 (Legacy)** to indicate that the partition to which the event belongs cannot be determined.

The following legacy events may appear after your upgrade.

- User authentication succeeded.

© Copyright IBM Corp. 1999, 2018

- User authentication failed.
- Authentication failed because a user attempted to log in with too many concurrent sessions.
- User logged off and the corresponding session ended.
- User's password changed.

Legacy audit events are visible in reports only when you access the report with an account that has the PlatformAdminRole role in Marketing Platform. The pre-defined platform_admin user has this role.

Retroactive changes

If the first name, last name, or email address of a user account is changed, all audit events tracked for this user reflect the changes. This is true even for events tracked before the user profile changes were made.

Permissions for viewing the Audit Events report in a multi-partition environment

In a multi-partition environment, the partition membership of the administrator viewing the Audit Events report affects the events that are included when the administrator views the report.

For all audit events except configuration events, the report shows only those events that occurred in the partition of the administrator viewing the report. Events that occurred in other partitions are not shown in the report.

The exception is administrators with the PlatformAdminRole role, who can see events that occur in all partitions.

All configuration events are visible to all administrators who can view the report.

Enabling and disabling event auditing

By default, event auditing is disabled. To enable event auditing, you set the **IBM Marketing Platform | Audit Events | Is Event Auditing enabled** configuration property to True.

This configuration property affects only the audit events listed under **IBM Marketing Platform | Audit Events** on the Configuration page. The events tracked in the system log are not affected.

You can disable event auditing at any time by setting the value of the **Is Event Auditing enabled** configuration property to False.

The Audit Events report does not include the events controlled by the **Is Event Auditing enabled** property that occurred during any period when the property was set to **False**.

Configuring which audit events appear in the report

To specify the audit events that are available in the audit report and their severity, you set properties on the **Settings > Configuration** page.

1. Go to the **Settings > Configuration** page and expand the **IBM Marketing Platform | Audit Events | Audit Events Configuration** category.
2. Select the events that you want to track.

   The tracked events are available for inclusion in the audit report.
3. Expand the **IBM Marketing Platform | Audit Events | Audit events severity configuration** category and click **Edit settings**.
4. Specify the severity level that you want to assign to each of the tracked events.

   Select from the following options.
   - No severity
   - Informational
   - Warning
   - Critical

   The specified severity appears in the audit report, and can be used in filtering the report.

If you want to track the user session event **Record login and logout events for members of the HighSeverityAccounts group**, add the users whose login and logout events you want to track to the **highSeverityAccounts** group. You do this on the **Settings > User groups** page.

This group is created automatically in the default partition during installation. In a multi-partition environment, this group is created automatically when you create a new partition using the Marketing Platform `partitionTool` utility.

Related tasks:

"Adding a user to a group or subgroup" on page 22

Related reference:

"IBM Marketing Platform | Audit Events" on page 274

"IBM Marketing Platform | Audit Events | Audit events configuration" on page 274

"IBM Marketing Platform | Audit Events | Audit events severity configuration" on page 277

Modifying the audit report content and display

You can add and remove events and columns, rearrange and sort the columns, set the time span, specify which tracked events are shown in the report, and filter the information.

When you open the audit report without setting any report parameters, the following default settings are used.

- All of the events selected on the **Settings > Configuration** page under the **IBM Marketing Platform | Audit Events | Audit Events Configuration** category are shown, on multiple pages if necessary.
- No date criteria are applied.
- Events are sorted as follows: Event Date/Time (Descending), Login Name(Ascending), Severity Level (Ascending)

Use this procedure to modify these settings.

1. Go to **Analytics > Marketing Platform**.

2. To change the content of the report, do the following.
   a. Click the **Report Parameters** button.

      The Report Parameters window opens.
   b. Complete the fields.

      To set the sort order in the report, you can select from pre-defined sort orders in this window. You can also click the column headers in the report to sort on those columns.
   c. Click **Next** to move to a page where you can select which events are shown in the report.
   d. Click **Save and Close** to apply your selections to the report.
3. To filter the report, enter text or numbers in the **Filter** field and click the **Filter** button.

   The report displays only those events that contain the filter characters in any of the columns displayed in the report.

   To clear the filter, click the **X** in the Filter field.

Related reference:

"Fields in the Report Parameters window"

Fields in the Report Parameters window

Use the fields on the Report Parameters page to configure the way the audit report is displayed.

Table 73. Fields in the Report Parameters window

Field	Description
Sort	Select a sort order from the drop-down menu. Various combinations of column sorting are listed, along with whether the sort is in descending or ascending order. You can also sort columns using controls on the report page.
Time Period	Select from pre-defined time periods in the drop-down list, or enter start and end dates for a custom range.
Events	Select the optional events that you want to include in the report. For an event to be available in the report, it must be selected in the **IBM Marketing Platform | Audit Events | Audit Events Configuration** category on the **Settings > Configuration** page.
Columns	Use the **Add** and **Remove** buttons to specify the optional columns you want to appear in the report.

Related tasks:

"Modifying the audit report content and display" on page 209

Fields and buttons in the Audit Events report

Fields in the Audit Events report provide details about system and user events in IBM Marketing Software.

Table 74. Fields and buttons in the Audit Event report

Field or button	Description
Filter	To filter the report, enter text or numbers in the Filter field and click the button. The report displays only those events that contain the filter characters in any of the columns displayed in the report.
Report parameters	Click to open a window where you can change the columns displayed in the report, set a time period, and select from pre-defined sort orders.
Export	Click to open a window where you can export the report in CSV or text format.
Refresh	Click to refresh the report data.
Default fields	
Event date/time (short)	The date and time of the event on the server where Marketing Platform is deployed.
Event name	The tracked event. Events that are tracked are specified on the **Settings > Configuration** page.
Event details	Details about the tracked event. When a link is present, you can click it to see full details.
Login name	The login name of the user who performed the action.
Last name, first name	The first and last name of the user who performed the action.
Severity	The severity assigned to the event on the **IBM Marketing Platform | Audit Events | Audit Events Severity Configuration** page.
Optional fields set in the Report Parameters window	
Browser	The browser used by the person who performed the action.
Host name	The name or IP address of the machine from which the action was performed.
Request from	The URI where the HTTP request originated.
Event date/time (long)	The date and time of the event on the server where Marketing Platform is deployed, including the time zone.
User email	The email address of the user who performed the action.
Partition	The partition membership of the user who performed the action.

Archived audit events

You can configure backups of audit events by setting the value of configuration properties in the **IBM Marketing Platform | Audit Events | Audit Events Configuration** category on the **Settings > Configuration** page.

The archived data is stored in the USM_AUDIT_BACKUP table and can be included in the Audit Events report when you set a custom date range that includes data from the archive. Loading a report that includes archived data can take longer than loading a report that does not include archived data.

The system posts a notification when an audit backup process completes. You can also configure a group of users who receive these notifications in emails.

Set the following properties to configure audit event backups.

- **Enable Audit Backup**
- **Archive data after the number of days specified here**
- **Keep Audit records in primary for number days specified here**
- **Archive start time**
- **Name of group to receive audit backup notifications**

Related tasks:

"Configuring audit backup notifications"

Related reference:

"IBM Marketing Platform | Audit Events | Audit events configuration" on page 274

Configuring audit backup notifications

To notify users of the status of audit event backup, make them members of a group that you specify in a configuration property.

1. Determine the group whose members you want to receive email notifications of audit data backups.

   You can use an existing group or create a new one on the **Settings > User groups** page.

   You can specify only a single group to receive notifications.
2. Go to the **Settings > Configuration** page and expand the **IBM Marketing Platform | Audit Events | Audit events configuration** category.
3. Set the value of the **Name of group to receive audit backup notifications** property to the name of the group you selected.
4. Add the users who should receive notifications to the group.
5. The users who you have added to the group must subscribe to the notifications on the **Settings > Users** page.

   An administrator can do this for each user, or you can inform the users that they must go to their account, click **Notification subscriptions**, and subscribe to **Audit backup** notifications.

Each time the system backs up audit data, an email notification and user interface notification is generated for the members of the group you specified, if they have subscribed to Audit Event notifications.

Related concepts:

"Archived audit events" on page 211

Exporting the Audit Events report

You can export the security audit report to a text or comma separated file.

1. Go to **Analytics > Marketing Platform**.
2. Click the **Export** button.
3. In the Audit Report Export window, enter a name for your exported report, and select the export format.

   The format options are:

   - **CSV** (a comma separated list that Microsoft Excel can open)
   - **TXT** (text)

     If you select text format, you also choose the separator. Options are:

     – **#**

     – **|**

     – **TAB**
4. Click **Export**, specify whether you want to open or save the exported report, and then close the export window.

Optimizing the export of the Audit Events report for large event volumes

If you want to export large audit event reports, for example, reports containing more than 65,000 records matching your audit event report filter criteria, the export can time out. To circumvent this problem, perform the following procedure.

This procedure applies when you use Internet Explorer to access the Audit Event report.

1. Edit the Windows registry as follows.
   a. Open the Windows registry editor and navigate to `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings`.
   b. If a DWORD entry named `ReceiveTimeout` does not exist, create one.

      To create a new DWORD entry, do the following.

      - Right-click on `Internet Settings` and select **New > DWORD (32-bit) Value**.
      - Enter `ReceiveTimeout` as the name for the new entry.

   c. Give the existing or new `ReceiveTimeout` entry a value as follows.

      - Right-click on the `ReceiveTimeout` entry and select **Modify**.
      - Under **Base**, select **Decimal**.
      - Specify the timeout interval in milliseconds.

        For example, to specify 3 hours, you would enter 10800000, which is 180 minutes * 60 seconds * 1000.
2. Configure Internet Explorer as follows.
   a. Select **Tools > Internet Options** and click the Security tab.
   b. Select the zone where you access Marketing Platform. For example, Trusted Sites.
   c. Click **Custom level**.
   d. Under **Downloads**, enable **Automatic prompting for file downloads**.
   e. Restart Internet Explorer.

Chapter 19. The Marketing Platform system log

You should check the system log first if the Marketing Platform application malfunctions. The system log is independent of the security audit information, which is stored in the system tables. While the system log tracks some of the same information contained in the security audit reports, it also contains information useful in troubleshooting Marketing Platform.

The system log contains the following information.

- Configuration information and all errors and debugging information for the Marketing Platform.
- A record of key events as they occur on the Marketing Platform server (requests, grants, revokes, and failures).

About the configuration settings displayed in the system log

Note: If a problem occurs when the system attempts to write to the system log file, the system writes to stdout (command line) instead of to a file.

System log entry format

The system log entries are in the following format.

```
Timestamp | Event severity level | Message
```

- **Timestamp** – The time the event occurred.
- **Event Severity Level** – The logging level of the event.
- **Message** – Description of the event. If the entry is a request to the server, the message typically contains the function called by the request. Response entries record the results of the requests.

System log configuration

You configure the system log using the `log4j.properties` file, located by default in the `conf` directory under your Marketing Platform installation. Changes to this file go into effect within 30 seconds after the file is saved.

Configuration you perform on the system log does not affect security audit reports.

Default system log settings

By default, the system log is configured as follows:

- Log file name: `platform.log`
- Log directory: `Unica/Platform/logs`
- Log level: `WARN`
- Number of backups: 1
- Maximum size of log files: 10MB

Note the following.

- If you increase the number of backups or size of the log files, verify that the machine on which the logs are stored has sufficient memory.

© Copyright IBM Corp. 1999, 2018

- Setting the logging level higher than the default might affect performance.

Logging levels in the system log

The possible logging levels in the system log are as follows, in ascending order.

- ERROR
- WARN
- INFO
- DEBUG
- TRACE

The higher levels include the information contained in all of the lower levels. For example, setting the level to DEBUG enables the DEBUG, INFO, WARN and ERROR traces.

If the logging level is set to DEBUG, the response messages include any SQL queries performed against the Marketing Platform data store.

Logging level settings for the whole Marketing Platform system

You can change the logging level for all components of Marketing Platform by uncommenting the desired line in the Examples section of the file. To uncomment a line, remove the # character at the beginning of the line. If you make this change, be sure to add the # symbol to the beginning of the line specifying the previous logging level.

Setting logging levels for Marketing Platform components

You can set the logging level in the system log for specific components of the Marketing Platform. These components include:

- Localization
- User and group processing
- Data migration
- LDAP integration
- Authentication (server-side processing)
- The Configuration pages
- Database access
- Various third-party libraries (for example, ibatis)

By default, the component-level logging is turned off. To debug a specific module, remove the # character at the start of each line of the module in the log4j.properties file.

Where to find more information about log4j

You can find additional information about log4j in the following ways.

- See comments in the log4j.properties file.
- See http://logging.apache.org/log4j/docs/documentation.html.

Enabling single-user logging

You can enable single-user logging by configuring logging to use the XML file and then editing the XML file.

Logging is configured using one of two files: `log4j.properties` or `log4j.xml`. By default, the `log4j.properties` file is used.

You can enable per-user logging by configuring logging to use the XML file and then editing the XML file. If Marketing Platform is configured in a cluster deployment, copy the XML file to each node.

Note: With XML logging enabled, a thread is created that periodically checks if the XML configuration file has been created or modified. If a change or file creation is detected, then the XML file is read to configure log4j. The polling interval is 60 seconds.

1. Configure logging to use `log4j.xml` by setting the following JVM parameter.

   `-DENABLE_PLATFORM_LOG4J_XML_LOGGING=true`

   The value must be set to `true` to enable per-user logging.

   If Marketing Platform is configured in a cluster deployment, set this JVM parameter in each node of the cluster.
2. To specify the user account to be logged in per-user logging, edit the `log4j.xml` file and add the users in the filter tag. The logs for the users that are added in the filter tag are saved in the file that is mentioned. You can also create an appender without the filter tag.
   - You can add multiple tags in the `log4j.xml` file to create separate log files for specific users. You must add a new appender for each new user specific log file.
   - By default, the log file is created in the `Platform_Home /Platform/logs` folder and is named as `platform.log`. You can specify a different valid path and file name. You must specify the absolute or complete path to generate the log files in the respective folders.
   - If both user specific logs and logs for all users are required, add an appender tag with a new name and without the filter tag defined. The appender must have a unique name.
   - Add a corresponding entry under the root tag for this new appender.
3. If Marketing Platform is configured in a cluster deployment, copy the edited XML file to each node of the cluster.

   You can use a command like the one shown in the following example.
   `-DPLATFORM_LOG4J_XML_FILE=log4j_node1.xml`

   The `log4j_node1.xml` file is a copy of the `log4j.xml` file. You can use any name for the copied file.

Consider the following example where the logs are collected for the user `asm_admin` and also for all other users.

```
<appender name="Console" class="org.apache.log4j.ConsoleAppender">
  <param name="ImmediateFlush" value="true"/>
  <layout class="org.apache.log4j.PatternLayout">
    <param name="ConversionPattern" value="%-5p %c - %m%n"/>
  </layout>
  <filter class="com.unica.manager.logger.UserMatchFilter">
    <param name="StringToMatch" value="asm_admin" />
  </filter>
</appender>

<!-- The following section is for user specific logs for the user asm_admin -->
<appender name="System" class="org.apache.log4j.RollingFileAppender">
  <param name="File" value="${UNICA_PLATFORM_LOG_FILE}"/>
  <!-- <param name="Encoding" value="utf-8"/>
  <param name="Append" value="true"/>
```

```
  <param name="ImmediateFlush" value="false"/>
  -->
  <param name="MaxBackupIndex" value="1"/>
  <param name="MaxFileSize" value="10MB"/>
  <layout class="org.apache.log4j.PatternLayout">
    <param name="ConversionPattern" value="%d{DATE} - %-5p - %m%n"/>
  </layout>
  <filter class="com.unica.manager.logger.UserMatchFilter">
    <param name="StringToMatch" value="asm_admin" />
  </filter>
</appender>

<!-- The following section is for logs for all the users -->
<appender name="SystemAllUsers" class="org.apache.log4j.RollingFileAppender">
  <param name="File" value="<LOG_FILE_PATH>"/><!--the absolute path for the log file-->
  <!-- <param name="Encoding" value="utf-8"/>
  <param name="Append" value="true"/>
  <param name="ImmediateFlush" value="false"/>
  -->
  <param name="MaxBackupIndex" value="1"/>
  <param name="MaxFileSize" value="10MB"/>
  <layout class="org.apache.log4j.PatternLayout">
    <param name="ConversionPattern" value="%d{DATE} - %-5p - %m%n"/>
  </layout>
</appender>

<!-- <logger name="com.unica.manager.configuration.ConfigurationManager">
    <level value="TRACE"/>
  </logger>
  <logger name="com.unica.suite.scheduler.server.manager.TaskManager">
    <level value="DEBUG"/>
  </logger>
  <logger name="org.hibernate.util.JDBCExceptionReporter">
    <level value="ERROR"/>
  </logger>
-->
<root>
  <level value="WARN"/>
    <appender-ref ref="System"/>
    <appender-ref ref="Console"/>
    <appender-ref ref="SystemAllUsers"/>
</root>
```

Chapter 20. IBM Marketing Platform utilities and SQL scripts

This section provides an overview of the Marketing Platform utilities, including some details that apply to all of the utilities and which are not included in the individual utility descriptions.

Location of utilities

Marketing Platform utilities are located in the `tools/bin` directory under your Marketing Platform installation.

List and descriptions of utilities

The Marketing Platform provides the following utilities.

- "alertConfigTool" on page 221 - registers alerts and configurations for IBM Marketing Software products
- "configTool" on page 222 - imports, exports, and deletes configuration settings, including product registrations
- "datafilteringScriptTool" on page 226 - creates data filters
- "encryptPasswords" on page 227 - encrypts and stores passwords
- "partitionTool" on page 228 - creates database entries for partitions
- "populateDb" on page 231 - populates the Marketing Platform database
- "restoreAccess" on page 231 - restores a user with the platformAdminRole role
- "scheduler_console_client" on page 233 - lists or starts IBM Marketing Software Scheduler jobs that are configured to listen for a trigger.

Prerequisites for running Marketing Platform utilities

The following are prerequisites for running all Marketing Platform utilities.

- Run all utilities from the directory where they are located (by default, the `tools/bin` directory under your Marketing Platform installation).
- On UNIX, the best practice is to run the utilities with the same user account that runs the application server on which Marketing Platform is deployed. If you run a utility with a different user account, adjust the permissions on the `platform.log` file to allow that user account to write to it. If you do not adjust permissions, the utility is not able to write to the log file and you might see some error messages, although the tool should still function correctly.

Authentication of utilities

Utilities such as `configTool` and other IBM Marketing Software back end utilities are designed to be used by system administrators and require physical access to the host servers for them to be invoked. For this reason, authentication for these utilities has been designed to be independent of the UI authentication mechanism. Access to these utilities is available to users with Marketing Platform administrator privileges. Access to these utilities is expected to be locally defined in Marketing Platform and authenticated against the same.

© Copyright IBM Corp. 1999, 2018

Troubleshooting connection issues

All of the Marketing Platform utilities except `encryptPasswords` interact with the Marketing Platform system tables. To connect to the system table database, these utilities use the following connection information, which is set by the installer using information provided when the Marketing Platform was installed. This information is stored in the `jdbc.properties` file, located in the `tools/bin` directory under your Marketing Platform installation.

- JDBC driver name
- JDBC connection URL (which includes the host, port, and database name)
- Data source login
- Data source password (encrypted)

In addition, these utilities rely on the `JAVA_HOME` environment variable, set either in the `setenv` script located in the `tools/bin` directory of your Marketing Platform installation, or on the command line. The Marketing Platform installer should have set this variable automatically in the `setenv` script, but it is a good practice to verify that the `JAVA_HOME` variable is set if you have a problem running a utility. The JDK must be the Sun version (not, for example, the JRockit JDK available with WebLogic).

Special characters

Characters that are designated as reserved characters in the operating system must be escaped. Consult your operating system documentation for a list of reserved characters and how to escape them.

Standard options in Marketing Platform utilities

The following options are available in all Marketing Platform utilities.

`-l logLevel`

Set the level of log information displayed in the console. Options are `high`, `medium`, and `low`. The default is `low`.

`-L`

Set the locale for console messages. The default locale is en_US. The available option values are determined by the languages into which the Marketing Platform has been translated. Specify the locale using the ICU locale ID according to ISO 639-1 and ISO 3166.

`-h`

Display a brief usage message in the console.

`-m`

Display the manual page for this utility in the console.

`-v`

Display more execution details in the console.

Setting up Marketing Platform utilities on additional machines

On the machine where the Marketing Platform is installed, you can run the Marketing Platform utilities without any additional configuration. However, you might want to run the utilities from another machine on the network. This procedure describes the steps required to do this.

Ensure that the machine on which you perform this procedure meets the following prerequisites.

- The correct JDBC driver must exist on the machine or be accessible from it.
- The machine must have network access to the Marketing Platform system tables.
- The Java runtime environment must be installed on the machine or be accessible from it.

1. Gather the following information about the Marketing Platform system tables.
   - The fully qualified path for the JDBC driver file or files on your system.
   - The fully qualified path to an installation of the Java runtime environment.

     The default value in the installer is the path to the supported version of the JRE that the installer places under your IBM Marketing Software installation directory. You can accept this default or specify a different path.
   - Database type
   - Database host
   - Database port
   - Database name/system ID
   - Database user name
   - Database password
2. Run the IBM Marketing Software installer and install the Marketing Platform.

   Enter the database connection information that you gathered for the Marketing Platform system tables. If you are not familiar with the IBM Marketing Software installer, see the Campaign or Marketing Operations installation guide.

   You do not have to deploy the Marketing Platform web application if you are installing the utilities only.

Marketing Platform utilities

This section describes the Marketing Platform utilities, with functional details, syntax, and examples.

alertConfigTool

Notification types are specific to the various IBM Marketing Software products. Use the `alertConfigTool` utility to register the notification types when the installer has not done this automatically during installation or upgrade.

Syntax

`alertConfigTool -i -f` *`importFile`*

Commands

-i -f *importFile*

Import alert and notification types from a specified XML file.

Example

- Import alert and notification types from a file named `Platform_alerts_configuration.xml` located in the `tools\bin` directory under the Marketing Platform installation.

  ```
  alertConfigTool -i -f Platform_alerts_configuration.xml
  ```

configTool

The properties and values on the Configuration page are stored in the Marketing Platform system tables. You can use the `configTool` utility to import and export configuration settings to and from the system tables.

When to use configTool

You might want to use `configTool` for the following reasons.

- To import partition and data source templates that are supplied with Campaign, which you can then modify and duplicate by using the Configuration page.
- To register (import configuration properties for) IBM Marketing Software products, if the product installer is unable to add the properties to the database automatically.
- To export an XML version of configuration settings for backup or to import into a different installation of IBM Marketing Software.
- To delete categories that do not have the **Delete Category** link. You do this by using `configTool` to export your configuration, then manually deleting the XML that creates the category, and by using `configTool` to import the edited XML.

Important: This utility modifies the `usm_configuration` and `usm_configuration_values` tables in the Marketing Platform system table database, which contains the configuration properties and their values. For best results, either create backup copies of these tables, or export your existing configurations by using `configTool` and back up the resulting file so you have a way to restore your configuration if you make an error when you use `configTool` to import.

Syntax

```
configTool -d -p "elementPath" [-o]

configTool -i -p "parent ElementPath" -f importFile [-o]

configTool -x -p "elementPath" -f exportFile

configTool -vp -p "elementPath" -f importFile [-d]

configTool -r productName -f registrationFile [-o] configTool -u
productName
```

Commands

-d -p "*elementPath*" [o]

Delete configuration properties and their settings, specifying a path in the configuration property hierarchy.

The element path must use the internal names of categories and properties. You can obtain them by going to the Configuration page, selecting the wanted category or property, and looking at the path that is displayed in parentheses in the right pane. Delimit a path in the configuration property hierarchy by using the | character, and surround the path with double quotation marks.

Note the following.

- Only categories and properties within an application can be deleted by using this command, not whole applications. Use the `-u` command to unregister a whole application.
- To delete categories that do not have the **Delete Category** link on the Configuration page, use the `-o` option.

When you use `-d` with the `-vp` command, the configTool deletes any child nodes in the path you specify if those nodes are not included in the XML file you specify.

`-i -p "`*`parentElementPath`*`" -f `*`importFile`*` [o]`

Import configuration properties and their settings from a specified XML file.

To import, you specify a path to the parent element under which you want to import your categories. The `configTool` utility imports properties under the category you specify in the path.

You can add categories at any level below the top level, but you cannot add a category at same level as the top category.

The parent element path must use the internal names of categories and properties. You can obtain them by going to the Configuration page, selecting the required category or property, and looking at the path that is displayed in parentheses in the right pane. Delimit a path in the configuration property hierarchy by using the | character, and surround the path with double quotation marks.

You can specify an import file location relative to the `tools/bin` directory or you can specify a full directory path. If you specify a relative path or no path, `configTool` first looks for the file relative to the `tools/bin` directory.

By default, this command does not overwrite an existing category, but you can use the `-o` option to force an overwrite.

`-x -p "`*`elementPath`*`" -f `*`exportFile`*

Export configuration properties and their settings to an XML file with a specified name.

You can export all configuration properties or limit the export to a specific category by specifying a path in the configuration property hierarchy.

The element path must use the internal names of categories and properties, which you can obtain by going to the Configuration page, selecting the wanted category or property, and looking at the path that is displayed in parentheses in the right

pane. Delimit a path in the configuration property hierarchy by using the | character, and surround the path with double quotation marks.

You can specify an export file location relative to the current directory or you can specify a full directory path. If the file specification does not contain a separator (/ on UNIX, / or \ on Windows), `configTool` writes the file to the `tools/bin` directory under your Marketing Platform installation. If you do not provide the `xml` extension, `configTool` adds it.

-vp -p "*elementPath*" -f *importFile* [-d]

This command is used mainly in manual upgrades, to import configuration properties. If you applied a fix pack that contains a new configuration property, and you then upgrade, importing a configuration file as part of a manual upgrade process can override values that were set when the fix pack was applied. The `-vp` command ensures that the import does not override previously set configuration values.

Important: After you use the `configTool` utility with the -vp option, you must restart the web application server on which Marketing Platform is deployed so the changes are applied.

When you use `-d` with the `-vp` command, the configTool deletes any child nodes in the path you specify if those nodes are not included in the XML file you specify.

-r *productName* -f *registrationFile*

Register the application. The registration file location can be relative to the `tools/bin` directory or can be a full path. By default, this command does not overwrite an existing configuration, but you can use the `-o` option to force an overwrite. The *productName* parameter must be one of those names that are listed above.

Note the following.

- When you use the `-r` command, the registration file must have `<application>` as the first tag in the XML.

  Other files can be provided with your product that you can use to insert configuration properties into the Marketing Platform database. For these files, use the `-i` command. Only the file that has the `<application>` tag as the first tag can be used with the `-r` command.
- The registration file for the Marketing Platform is named `Manager_config.xml`, and the first tag is `<Suite>`. To register this file on a new installation, use the `populateDb` utility, or rerun the Marketing Platform installer as described in the *IBM Marketing Platform Installation Guide*.
- After the initial installation, to re-register products other than the Marketing Platform, use `configTool` with the `-r` command and `-o` to overwrite the existing properties.

The `configTool` utility uses product names as parameters with the commands that register and unregister products. With the 8.5.0 release of IBM Marketing Software, many product names changed. However, the names that are recognized by `configTool` did not change. The valid product names for use with `configTool` are listed below, along with the current names of the products.

Table 75. Product names for configTool registration and unregistration

Product name	Name used in configTool
Marketing Platform	`Manager`
Campaign	`Campaign`
Distributed Marketing	`Collaborate`
eMessage	`emessage`
Interact	`interact`
Contact Optimization	`Optimize`
Marketing Operations	`Plan`
Opportunity Detect	`Detect`
IBM SPSS Modeler Advantage Enterprise Marketing Management Edition	`SPSS`
Digital Analytics	`Coremetrics`

-u *productName*

Unregister an application that is specified by *productName*. You do not have to include a path to the product category; the product name is sufficient, and it is required. The process removes all properties and configuration settings for the product.

Options

-o

When used with `-i` or `-r`, it overwrites an existing category or product registration (node).

When used with `-d`, you can delete a category (node) that does not have the **Delete Category** link on the Configuration page.

Examples

- Import configuration settings from a file named `Product_config.xml` in the `conf` directory under the Marketing Platform installation.

  ```
  configTool -i -p "Affinium" -f Product_config.xml
  ```

- Import one of the supplied Campaign data source templates into the default Campaign partition, partition1. The example assumes that you placed the Oracle data source template, `OracleTemplate.xml`, in the `tools/bin` directory under the Marketing Platform installation.

  ```
  configTool -i -p "Affinium|Campaign|partitions|partition1|dataSources" -f
  OracleTemplate.xml
  ```

- Export all configuration settings to a file named `myConfig.xml` in the `D:\backups` directory.

  ```
  configTool -x -f D:\backups\myConfig.xml
  ```

- Export an existing Campaign partition (complete with data source entries), save it to a file named `partitionTemplate.xml`, and store it in the default `tools/bin` directory under the Marketing Platform installation.

  ```
  configTool -x -p "Affinium|Campaign|partitions|partition1" -f
  partitionTemplate.xml
  ```

- Manually register an application named productName, by using a file named app_config.xml in the default tools/bin directory under the Marketing Platform installation, and force it to overwrite an existing registration of this application.

  ```
  configTool -r product Name -f app_config.xml -o
  ```

- Unregister an application named productName.

  ```
  configTool -u productName
  ```

Related concepts:

"Configuration management" on page 3

Related reference:

"ManagerSchema_DeleteAll.sql" on page 235

"ManagerSchema_DropAll.sql" on page 236

datafilteringScriptTool

The datafilteringScriptTool utility reads an XML file to populate the data filtering tables in the Marketing Platform system table database.

Depending on how you write the XML, you can use this utility in two ways.

- Using one set of XML elements, you can auto-generate data filters based on unique combinations of field values (one data filter for each unique combination).
- Using a slightly different set of XML elements, you can specify each data filter that the utility creates.

See *IBM Marketing Platform the Administrator's Guide* for information about creating the XML.

When to use datafilteringScriptTool

You must use datafilteringScriptTool when you create new data filters.

Prerequisites

The Marketing Platform must be deployed and running.

Using datafilteringScriptTool with SSL

When the Marketing Platform is deployed using one-way SSL you must modify the datafilteringScriptTool script to add the SSL options that perform handshaking. To modify the script, you must have the following information.

- Truststore file name and path
- Truststore password

In a text editor, open the datafilteringScriptTool script (.bat or .sh) and find the lines that look like this (examples are Windows version).

```
:callexec

"%JAVA_HOME%\bin\java" -DUNICA_PLATFORM_HOME="%UNICA_PLATFORM_HOME%"

com.unica.management.client.datafiltering.tool.DataFilteringScriptTool %*
```

Edit these lines to look like this (new text is in **bold**). Substitute your truststore path and file name and truststore password for `myTrustStore.jks` and `myPassword`.

```
:callexec

SET SSL_OPTIONS=-Djavax.net.ssl.keyStoreType="JKS"

-Djavax.net.ssl.trustStore="C:\security\myTrustStore.jks"

-Djavax.net.ssl.trustStorePassword=myPassword

"%JAVA_HOME%\bin\java" -DUNICA_PLATFORM_HOME="%UNICA_PLATFORM_HOME%"
%SSL_OPTIONS%

com.unica.management.client.datafiltering.tool.DataFilteringScriptTool %*
```

Syntax

```
datafilteringScriptTool -r pathfile
```

Commands

-r *path*_file

Import data filter specifications from a specified XML file. If the file is not located in the `tools/bin` directory under your installation, provide a path and enclose the *path*_file parameter in double quotation marks.

Example

- Use a file named `collaborateDataFilters.xml`, located in the `C:\unica\xml` directory, to populate the data filter system tables.

  ```
  datafilteringScriptTool -r "C:\unica\xml\collaborateDataFilters.xml"
  ```

Related tasks:

"Populating the data filter system tables" on page 183

encryptPasswords

The `encryptPasswords` utility is used to encrypt and store either of two passwords that Marketing Platform uses internally.

The two passwords that the utility can encrypt are as follows.

- The password that the Marketing Platform uses to access its system tables. The utility replaces an existing encrypted password (stored in the `jdbc,properties` file, located in the `tools\bin` directory under your Marketing Platform installation) with a new one.
- The keystore password used by the Marketing Platform when it is configured to use SSL with a certificate other than the default one supplied with the Marketing Platform or the web application server. The certificate can be either a self-signed certificate or a certificate from a certificate authority.

When to use encryptPasswords

Use `encryptPasswords` as for the following reasons.

- When you change the password of the account used to access your Marketing Platform system table database.
- When you have created a self-signed certificate or have obtained one from a certificate authority.

Prerequisites

- Before running `encryptPasswords` to encrypt and store a new database password, make a backup copy of the `jdbc.properties` file, located in the `tools/bin` directory under your Marketing Platform installation.
- Before running `encryptPasswords` to encrypt and store the keystore password, you must have created or obtained a digital certificate and know the keystore password.

Syntax

```
encryptPasswords -d databasePassword

encryptPasswords -k keystorePassword
```

Commands

`-d databasePassword`

Encrypt the database password.

`-k keystorePassword`

Encrypt the keystore password and store it in a file named `pfile`.

Examples

- When the Marketing Platformwas installed, the login for the system table database account was set to `myLogin`. Now, some time after installation, you have changed the password for this account to `newPassword`. Run `encryptPasswords` as follows to encrypt and store the database password.

  ```
  encryptPasswords -d newPassword
  ```

- You are configuring an IBM Marketing Software application to use SSL and have created or obtained a digital certificate. Run `encryptPasswords` as follows to encrypt and store the keystore password.

  ```
  encryptPasswords -k myPassword
  ```

Related concepts:

Chapter 20, “IBM Marketing Platform utilities and SQL scripts,” on page 219

partitionTool

Partitions are associated with Campaign policies and roles. These policies and roles and their partition associations are stored in the Marketing Platform system tables. The `partitionTool` utility seeds the Marketing Platform system tables with basic policy and role information for partitions.

When to use partitionTool

For each partition you create, you must use `partitionTool` to seed the Marketing Platform system tables with basic policy and role information.

See the installation guide appropriate for your version of Campaign for detailed instructions on setting up multiple partitions in Campaign.

Special characters and spaces

Any partition description or user, group, or partition name that contains spaces must be enclosed in double quotation marks.

Syntax

`partitionTool -c -s` *`sourcePartition`* `-n` *`newPartitionName`* `[-u` *`admin_user_name`*`] [-d` *`partitionDescription`*`] [-g` *`groupName`*`]`

Commands

The following commands are available in the `partitionTool` utility.

`-c`

Replicates (clones) the policies and roles for an existing partition specified using the `-s` option, and uses the name specified using the `-n` option. Both of these options are required with `c`. This command does the following.

- Creates a new IBM Marketing Software user with the Admin role in both the Administrative Roles policy and the global policy in Campaign. The partition name you specify is automatically set as this user's password.
- Creates a new Marketing Platform group and makes the new Admin user a member of that group.
- Creates a new partition object.
- Replicates all the policies associated with the source partition and associates them with the new partition.
- For each replicated policy, replicates all roles associated with the policy.
- For each replicated role, maps all functions in the same way that they were mapped in the source role.
- Assigns the new Marketing Platform group to the last system-defined Admin role created during role replication. If you are cloning the default partition, partition1, this role is the default Administrative Role (Admin).

Options

`-d` ***`partitionDescription`***

Optional, used with `-c` only. Specifies a description that appears in the output from the `-list` command. Must be 256 characters or less. Enclose in double quotation marks if the description contains spaces.

`-g` ***`groupName`***

Optional, used with -c only. Specifies the name of the Marketing Platform Admin group that the utility creates. The name must be unique within this instance of Marketing Platform

If not defined, the name defaults to partition_nameAdminGroup.

-n *partitionName*

Optional with -list, required with -c. Must be 32 characters or less.

When used with -list, specifies the partition whose information is listed.

When used with -c, specifies the name of the new partition, and the partition name you specify is used as the password for the Admin user. The partition name must match the name you gave the partition in when you configured it (using the partition template on the Configuration page).

-s *sourcePartition*

Required, used with -c only. The name of the source partition to be replicated.

-u *adminUserName*

Optional, used with -c only. Specifies the user name of the Admin user for the replicated partition. The name must be unique within this instance of Marketing Platform.

If not defined, the name defaults to *partitionName*AdminUser.

The partition name is automatically set as this user's password.

Examples

- Create a partition with the following characteristics.
  - Cloned from partition1
  - Partition name is myPartition
  - Uses the default user name (myPartitionAdminUser) and password (myPartition)
  - Uses the default group name (myPartitionAdminGroup)
  - Description is "ClonedFromPartition1"

  ```
  partitionTool -c -s partition1 -n myPartition -d "ClonedFromPartition1"
  ```
- Create a partition with the following characteristics.
  - Cloned from partition1
  - Partition name is partition2
  - Specifies user name of customerA with the automatically assigned password of partition2
  - Specifies group name of customerAGroup
  - Description is "PartitionForCustomerAGroup"

  ```
  partitionTool -c -s partition1 -n partition2 -u customerA -g
  customerAGroup -d "PartitionForCustomerAGroup"
  ```

Related concepts:

Chapter 20, “IBM Marketing Platform utilities and SQL scripts,” on page 219

populateDb

The `populateDb` utility inserts default (seed) data in the Marketing Platform system tables.

The IBM Marketing Software installer can populate the Marketing Platform system tables with default data for Marketing Platform and for Campaign. However, if your company policy does not permit the installer to change the database, or if the installer is unable to connect with the Marketing Platform system tables, you must insert default data in the Marketing Platform system tables using this utility.

For Campaign, this data includes security roles and permissions for the default partition. For Marketing Platform, this data includes default users and groups, and security roles and permissions for the default partition.

Syntax

`populateDb -n productName`

Commands

`-n productName`

Insert default data into the Marketing Platform system tables. Valid product names are `Manager` (for Marketing Platform) and `Campaign` (for Campaign).

Examples

- Insert Marketing Platform default data manually.

  ```
  populateDb -n Manager
  ```

- Insert Campaign default data manually.

  ```
  populateDb -n Campaign
  ```

Related reference:

"ManagerSchema_DeleteAll.sql" on page 235

"ManagerSchema_DropAll.sql" on page 236

restoreAccess

The `restoreAccess` utility allows you to restore access to Marketing Platform if all users with PlatformAdminRole privileges have been inadvertently locked out or if all ability to log in to the Marketing Platform has been lost.

When to use restoreAccess

You might want to use `restoreAccess` under the two circumstances described in this section.

PlatformAdminRole users disabled

It is possible that all users with PlatformAdminRole privileges in Marketing Platform might become disabled in the system. Here is an example of how the platform_admin user account might become disabled. Suppose you have only one

user with PlatformAdminRole privileges (the platform_admin user). Assume the `Maximum failed login attempts allowed property` property in the **General | Password settings** category on the Configuration page is set to 3. Then suppose someone who is attempting to log in as platform_admin enters an incorrect password three times in a row. These failed login attempts cause the platform_admin account to become disabled in the system.

In that case, you can use `restoreAccess` to add a user with PlatformAdminRole privileges to the Marketing Platform system tables without accessing the web interface.

When you run `restoreAccess` in this way, the utility creates a user with the login name and password you specify, and with PlatformAdminRole privileges.

If the user login name you specify exists in Marketing Platform as an internal user, that user's password is changed.

Only a user with the login name of PlatformAdmin and with PlatformAdminRole privileges can universally administer all dashboards. So if the platform_admin user is disabled and you create a user with `restoreAccess`, you should create a user with a login of platform_admin.

Improper configuration of NTLMv2 authentication

If you implement NTLMv2 authentication with improper configuration and can no longer log in, use `restoreAccess` to restore the ability to log in.

When you run `restoreAccess` in this way, the utility changes the value of the `Platform | Security | Login method` property to `Marketing Platform`. This change allows you to log in with any user account that existed before you were locked out. You can optionally specify a new login name and password as well. You must restart the web application server on which Marketing Platform is deployed if you use the `restoreAccess` utility in this way.

Password considerations

Note the following about passwords when you use `restoreAccess`.

- The `restoreAccess` utility does not support blank passwords, and does not enforce password rules.
- If you specify a user name that is in use, the utility resets the password for that user.

Syntax

```
restoreAccess -u loginName -p password

restoreAccess -r
```

Commands

`-r`

When used without the `-u` *`loginName`* option, reset the value of the `Platform | Security | Login method` property to `Marketing Platform`. Requires restart of the web application server to take effect.

When used with the **-u** *loginName* option, create a PlatformAdminRole user.

Options

-u ***loginNname***

Create a user with PlatformAdminRole privileges with the specified login name. Must be used with the **-p** option.

-p ***password***

Specify the password for the user being created. Required with **-u**.

Examples

- Create a user with PlatformAdminRole privileges. The login name is `tempUser` and the password is `tempPassword`.

  ```
  restoreAccess -u tempUser -p tempPassword
  ```

- Change the value of the login method to `IBM Marketing Platform` and create a user with PlatformAdminRole privileges. The login name is `tempUser` and the password is `tempPassword`.

  ```
  restoreAccess -r -u tempUser -p tempPassword
  ```

Related concepts:

"Cross-partition administration privileges" on page 20

scheduler_console_client

Jobs configured in the IBM Marketing Software Scheduler can be listed and kicked off by this utility, if they are set up to listen for a trigger.

What to do if SSL is enabled

When the Marketing Platform web application is configured to use SSL, the JVM used by the `scheduler_console_client` utility must use the same SSL certificate that is used by the web application server on which the Marketing Platform is deployed.

Take the following steps to import the SSL certificate

- Determine the location of the JRE used by the `scheduler_console_client`.
  - If `JAVA_HOME` is set as a system environment variable, the JRE it points to is the one used by the `scheduler_console_client` utility.
  - If `JAVA_HOME` is not set as a system environment variable, the `scheduler_console_client` utility uses the JRE set either in the `setenv` script located in the `tools/bin` directory of your Marketing Platform installation, or on the command line.
- Import the SSL certificate used by the web application server on which the Marketing Platform is deployed to the JRE used by `scheduler_console_client`.

  The Sun JDK includes a program called `keytool` that you can use to import the certificate. Consult the Java documentation for complete details on using this program, or access the help by entering `-help` when you run the program.

- Open the `tools/bin/schedulerconsoleclient` file in a text editor and add the following properties. These differ depending on the web application server on which Marketing Platform is deployed.
  - For WebSphere, add these properties to the file.

    -Djavax.net.ssl.keyStoreType=JKS

    -Djavax.net.ssl.keyStore="*Path to your key store JKS file*"

    -Djavax.net.ssl.keyStorePassword="*Your key store password*"

    -Djavax.net.ssl.trustStore="*Path to your trust store JKS file*"

    -Djavax.net.ssl.trustStorePassword="*Your trust store password*"

    -DisUseIBMSSLSocketFactory=false
  - For WebLogic, add these properties to the file.

    -Djavax.net.ssl.keyStoreType="JKS"

    -Djavax.net.ssl.trustStore="*Path to your trust store JKS file*"

    -Djavax.net.ssl.trustStorePassword="*Your trust store password*"

If the certificates do not match, the Marketing Platform log file contains an error such as the following.

```
Caused by: sun.security.provider.certpath.SunCertPathBuilderException:
unable to find valid certification path to requested target
```

Prerequisites

The Marketing Platform must be installed, deployed, and running.

Syntax

```
scheduler_console_client -v -t trigger_name user_name
```

```
scheduler_console_client -s -t trigger_name user_name
```

Commands

-v

List the scheduler jobs configured to listen for the specified trigger.

Must be used with the `-t` option.

-s

Send the specified trigger.

Must be used with the `-t` option.

Options

-t *trigger_name*

The name of the trigger, as configured in the scheduler.

Example

- List jobs configured to listen for a trigger named `trigger1`.

  ```
  scheduler_console_client -v -t trigger1 myLogin
  ```

- Execute jobs configured to listen for a trigger named `trigger1`.

  ```
  scheduler_console_client -s -t trigger1 myLogin
  ```

Related concepts:

"Schedule triggers that are sent from an external script" on page 63

Marketing Platform SQL scripts

This section describes the SQL scripts provided with Marketing Platform to perform various tasks relating to the Marketing Platform system tables.

The Marketing Platform SQL scripts are located in the `db` directory under your Marketing Platform installation.

The scripts are designed to be run against the Marketing Platform system tables, using the database client.

ManagerSchema_DeleteAll.sql

The `Manager_Schema_DeleteAll.sql` script removes all data from the Marketing Platform system tables without removing the tables themselves. This script removes all users, groups, security credentials, data filters, and configuration settings from Marketing Platform.

When to use ManagerSchema_DeleteAll.sql

You might want to use `ManagerSchema_DeleteAll.sql` if corrupted data prevents you from using an instance of Marketing Platform.

Additional requirements

To make Marketing Platform operational after running `ManagerSchema_DeleteAll.sql` , you must perform the following steps.

- Run the `populateDB` utility. The `populateDB` utility restores the default configuration properties, users, roles, and groups, but does not restore any users, roles, and groups you have created or imported after initial installation.
- Use the configTool utility with the `config_navigation.xml` file to import menu items.
- If you have performed any post-installation configuration, such as creating data filters or integrating with an LDAP server or web access control platform, you must perform these configurations again.
- If you want to restore previously existing data filters, run the `datafilteringScriptTool` utility using the XML originally created to specify the data filters.

Related reference:

"populateDb" on page 231

"configTool" on page 222

ManagerSchema_PurgeDataFiltering.sql

The ManagerSchema_PurgeDataFiltering.sql script removes all data filtering data from the Marketing Platform system tables without removing the data filter tables themselves. This script removes all data filters, data filter configurations, audiences, and data filter assignments from Marketing Platform.

When to use ManagerSchema_PurgeDataFiltering.sql

You might want to use ManagerSchema_PurgeDataFiltering.sql if you need to remove all data filters without removing other data in the Marketing Platform system tables.

Important: The ManagerSchema_PurgeDataFiltering.sql script does not reset the values of the two data filter properties, Default table name and Default audience name. If these values are no longer valid for the data filters you want to use, you must set the values manually on the Configuration page.

Related tasks:

"Populating the data filter system tables" on page 183

ManagerSchema_DropAll.sql

The ManagerSchema_DropAll.sql script removes all Marketing Platform system tables from a database. This script removes all tables, users, groups, security credentials, and configuration settings from Marketing Platform.

Note: If you run this script against a database containing an earlier version of the Marketing Platform system tables, you might receive error messages in your database client stating that constraints do not exist. You can safely ignore these messages.

When to use ManagerSchema_DropAll.sql

You might want to use ManagerSchema_DropAll.sql if you have uninstalled an instance of Marketing Platform where the system tables are in a database that contains other tables you want to continue using.

Additional requirements

To make the Marketing Platform operational after running this script, you must perform the following steps.

- Run the appropriate SQL script to re-create the system tables.
- Run the populateDB utility. Running the populateDB utility restores the default configuration properties, users, roles, and groups, but does not restore any users, roles, and groups you have created or imported after initial installation.
- Use the configTool utility with the config_navigation.xml file to import menu items.
- If you have performed any post-installation configuration, such as creating data filters or integrating with an LDAP server or web access control platform, you must perform these configurations again.

Related reference:

"SQL scripts for creating system tables" on page 237

"populateDb" on page 231

"configTool" on page 222

SQL scripts for creating system tables

Use the scripts described in the following table to create Marketing Platform system tables manually, when your company policy does not allow you to use the installer to create them automatically.

The scripts are shown in the order in which you must run them.

Table 76. Scripts for creating system tables

Datasource Type	Script Names
IBM DB2	• ManagerSchema_DB2.sql If you plan to support multi-byte characters (for example, Chinese, Japanese, or Korean), use the ManagerSchema_DB2_unicode.sql script. • ManagerSchema__DB2_CeateFKConstraints.sql • active_portlets.sql
Microsoft SQL Server	• ManagerSchema_SqlServer.sql • ManagerSchema__SqlServer_CeateFKConstraints.sql • active_portlets.sql
Oracle	• ManagerSchema_Oracle.sql • ManagerSchema__Oracle_CeateFKConstraints.sql • active_portlets.sql

If you plan to use the scheduler feature that enables you to configure a flowchart to run at predefined intervals, you must also create the tables that support this feature. To create the scheduler tables, run the appropriate script, as described in the following table.

Table 77. Scripts for enabling the IBM Marketing Software Scheduler

Data Source Type	Script Name
IBM DB2	quartz_db2.sql
Microsoft SQL Server	quartz_sqlServer.sql
Oracle	quartz_oracle.sql

When to use the create system tables scripts

You must use these scripts when you install or upgrade Marketing Platform if you have not allowed the installer to create the system tables automatically, or if you have used ManagerSchema_DropAll.sql to delete all Marketing Platform system tables from your database.

Related reference:

"ManagerSchema_DropAll.sql" on page 236

Chapter 21. Configuration properties on the Configuration page

This section describes the configuration properties found on the **Settings & Configuration** page.

Related concepts:

Chapter 4, “Configuration management,” on page 33

IBM Marketing Platform configuration properties

This section describes the Marketing Platform configuration properties on the Configuration page.

IBM Marketing Platform

Properties in this category allow you to set the default locale, and to set flags for whether the installation of Marketing Platform is clustered, whether Marketing Operations is integrated with Campaign, and whether offer integration is enabled for the integration.

Region

Description

Specifies the locale preference for IBM Marketing Software users. When you set this property on the Configuration page, the setting you apply is the default setting throughout IBM Marketing Software for all users, except those whose locale preference is set individually through the Marketing Platform User page. When you set this property for an individual user, the setting you apply for that user overrides the default setting.

This preference setting affects display of the language, time, numbers, and dates in IBM Marketing Software applications.

Availability of locales may vary depending on the IBM Marketing Software application, and not all IBM applications support this locale setting in the Marketing Platform. See specific product documentation to determine availability and support for the `Region setting` property.

Default value

`English (United States)`

Help server

Description

The URL of the server on which IBM hosted online help is installed. If IBM Marketing Software users have internet access, you should not change the default value, which points to the online help server maintained and updated by IBM .

Default value

The URL of the hosted help server.

Valid Values

© Copyright IBM Corp. 1999, 2018

Any server on which IBM hosted help is installed.

IBM Marketing Operations - Campaign integration

Description

A flag indicating whether Marketing Operations and Campaign are installed together and integrated. For more information about configuring this integration, see the *IBM Marketing Operations and Campaign Integration Guide*.

Default value

`False`

Valid Values

`True | False`

IBM Marketing Operations - Offer integration

Description

For systems the integrate Marketing Operations with Campaign, this flag indicates whether offer integration is also enabled. Offer integration enables the ability to use Marketing Operations to perform offer lifecycle management tasks. For more information about configuring this integration, see the *IBM Marketing Operations and Campaign Integration Guide*.

Default value

`False`

Valid Values

`True | False`

Start page

Description

The URL of the page that appears when users log in to IBM Marketing Software. The default is the default dashboard.

Default value

The default dashboard.

Valid Values

Any IBM Marketing Software URL except form submissions pages, edit pages, and search result pages.

Domain name

Description

The name of the domain where IBM Marketing Software is installed. The value is set during installation. You should not change this unless the domain name changes.

If users access IBM Marketing Software products with the Chrome browser, use the fully qualified domain name (FQDN). If the FQDN is not used, the Chrome browser cannot access the product URLs.

Default value

Not defined

Disable page tagging

Description

When set to the default value of `False`, IBM uses the Site ID code that was entered during Marketing Platform installation to gather basic statistics that track overall product usage trends to develop and improve IBM products. IBM sends the information to `http://pt200201.unica.com` over HTTP.

If you do not want to have such information collected, set this property to `True`.

Default value

`False`

Valid values

`True | False`

Is this deployment clustered

Description

If you install Marketing Platform in a clustered deployment, set this property to `True`. Otherwise, retain the default value of `False`.

If you change this property while Marketing Platform is running, you must restart Marketing Platform for the changes to take effect.

Default value

`False`

Valid values

`True | False`

Apply security on static content for all applications

Description

When this value is set to `Yes`, if an authenticated user attempts to directly access any static content such as an image, a check is performed to verify the user's authentication. If the user is authenticated, the content is rendered. If the user is not authenticated, the user is sent to the login page. This setting applies across all IBM Marketing Software products.

Default value

`No`

Valid values

`Yes | No`

IBM Marketing Software | General | Navigation

Properties in this category specify values that are used internally to navigate among IBM Marketing Software products.

TCP port for secure connections

Description

Specifies the SSL port in the web application server on which the Marketing Platform is deployed. This property is used internally for communication among IBM Marketing Software products.

Default value

7001

TCP port for standard connections

Description

Specifies the HTTP port in the web application server on which the Marketing Platform is deployed. This property is used internally for communication among IBM Marketing Software products.

Default value

7001

IBM Marketing Platform URL

Description

Specifies the URL used for Marketing Platform. This is set at installation time and normally should not be changed. Note that the URL contains the domain name, as shown in the following example.

```
protocol://machine_name_or_IP_address.domain_name:port_number/
context-root
```

The machine name should not be `localhost`.

If users access IBM Marketing Software products with the Chrome browser, use the fully qualified domain name (FQDN) in the URL. If the FQDN is not used, the Chrome browser cannot access the product URLs.

Important: If IBM Marketing Software products are installed in a distributed environment, you must use the machine name rather than an IP address in the navigation URL for all of the applications in the suite. Also, if you are on a clustered environment and choose to use ports that are different from the default ports `80` or `443` for your deployment, do not use a port number in the value of this property.

Default value

Not defined

Example

In an environment configured for SSL, the URL might look like this:

```
https://machineName.companyDomain.com:8080/unica
```

Related tasks:

"Setting web access control connection properties in IBM Marketing Software" on page 147

IBM Marketing Software | General | Data filtering

Properties in this category specify values used when data filtering is implemented.

Default table name

Description

This configuration property is required to enable data filters.

Set the value of this property to exactly match the name used for the `addTables | AddDataTable | dataTable | name` element in the XML used to create the data filters.

Default value

Undefined

Valid values

Maximum of 50 characters of type varchar.

Default audience name

Description

This configuration property is required to enable data filters.

Set the value of this property to exactly match the name used for the `AddAudience | audience | name` element in the XML used to create the data filters.

Default value

Undefined

Valid values

Maximum of 50 characters of type varchar.

Enable data filter cache

Description

This property is optional, and can be set to improve data filter performance.

This property specifies whether the Marketing Platform retrieves data filter definitions from the database or from a cache. When this value is **true**, data filter definitions are stored in the cache and the cache is updated whenever there is any change in the data filter definitions.

You must restart the Marketing Platform web application after you make a change in this property value before it can take effect.

Default value

`False`

Related concepts:

"Optional configuration property to improve data filter performance" on page 182

Related tasks:

"Setting required data filter configuration properties" on page 182

IBM Marketing Software | General | Password settings

Properties in **General | Password Settings** category specify the policies that apply to IBM Marketing Software passwords. Most of these password options apply only to passwords for internal users (created within the Marketing Platform), not to external users that are imported from an external system.

The exception is the `Maximum failed login attempts allowed` property, which affects both internal and external users. Also note that this property does not override any similar restriction set in an external system.

Maximum failed login attempts allowed

Description

Specifies the maximum number of times an invalid password may be entered each time a user logs in. If the maximum is reached, the user is disabled in the IBM Marketing Software system, and no one can log in as that user.

If set to zero or less, the system allows an infinite number of consecutive failures.

Default value

`3`

Valid values

Any integer

Password history count

Description

Specifies the number of old passwords the system retains for a user. The user is not allowed to reuse any passwords within this list of old passwords. If the value is set to zero or less, then no history is retained, and the user may reuse the same password repeatedly. Note that the password history count does not include the password initially assigned to a user account when it is created.

Default value

`0`

Valid values

Any integer

Validity (in days)

Description

Specifies the number of days before a user's password expires.

If the value is zero or less, then the password never expires.

If the value is greater than zero, users are required to change their password the first time they log in, and the expiration interval is counted from the date of the first login.

If you change this value after users and passwords have been created, the new expiration date takes effect for existing users the next time they change their password.

Default value

`30`

Valid values

Any integer

Blank passwords allowed

Description

Specifies whether the a blank password is allowed.If you set this to true you should also set `Minimum character length=0`.

Default value

`true`

Valid values

`true | false`

Allow identical user name and password

Description

Specifies whether the user's password is allowed to be the same as the user's login name.

Default value

`false`

Valid values

`true | false`

Minimum number of numeric characters

Description

Specifies the minimum number of numbers required in a password. If the value is zero or less, then there is no minimum requirement.

Default value

`0`

Valid values

Any integer

Minimum number of letter characters

Description

Specifies the minimum number of letters required in a password. If the value is zero or less, then there is no minimum requirement.

Default value

`0`

Valid values

Any integer

Minimum character length

Description

Specifies the minimum length of a password. If the value is zero or less, then there is no minimum requirement. If you set the value to greater than `0`, you should also set `Blank passwords allowed=false`.

Default value

4

Valid values

Any integer

IBM Marketing Software | General | Miscellaneous

Properties in this category specify values that are used internally, as well as a value you may need to set for the locale.

Token lifetime

Description

Specifies the length of time, in seconds, that a token generated by the Marketing Platform is valid. It is part of the suite sign-on implementation, and you should not change this value.

Default value

15

Valid values

Any positive integer

Default language

Description

Specifies the default language for the Marketing Platform. If you plan to install Campaign, you should set this value to match the locale set for Campaign in the `defaultLocale` property for Campaign.

Default value

English

Valid values

Supported locales

IBM Marketing Software | General | Communication | Email

Properties in this category are used to configure the Marketing Platform to send emails to users for system alerts and notifications.

Enable email communication

Description

When set to `True`, the Marketing Platform attempts to send emails to users for system alerts and notifications. The other properties in this category must also be set to enable this feature.

Default value

False

Email server protocol

Description

Specifies the protocol on the mail server that is used for sending system alerts and notifications to users. This is required for email notifications.

Default value

smtp

Email server host

Description

Specifies the name of the mail server used for sending system alerts and notifications to users. This is required for email notifications.

Default value

localhost

Email server port

Description

Specifies the port of the mail server used for sending system alerts and notifications to users. This is required for email notifications.

Default value

25

'From' address for emails

Description

Specifies the account from which system alert and notification emails are sent. If authentication is required on your mail server, use the email address of the account that you used when you saved a mail server account name and password as a data source in a Marketing Platform user account. This is required for email notifications.

Default value

Not defined

Authentication required for mail server?

Description

Specifies whether the mail server requires authentication.

Default value

False

IBM Marketing Software user for email account

Description

Specifies the user name of the Marketing Platform account where the email credentials are stored as a data source.

Required for notifications, only if your mail server requires authentication.

Default value

asm_admin

Data source for email account

Description

Specifies the name of the data source in the Marketing Platform account where the email credentials are stored.

Required for notifications, only if your mail server requires authentication.

Default value

`emailDS`

Related tasks:

"Configuring email notifications in IBM Marketing Software" on page 152

IBM Marketing Platform | Scheduler

Properties in this category allow you to enable and tune the performance of the IBM Marketing Software Scheduler.

Client polling interval (ms)

Configuration category

`Platform|Scheduler`

Description

IBM Campaign polls the IBM Marketing Software Scheduler for jobs at regular intervals, specified in milliseconds by this value. The default value is 60 seconds. Avoid setting this property to any value less than 10000 (10 seconds), because doing so can decrease campaign performance.

Default value

`60000`

Client initialization delay (ms)

Description

The amount of time, expressed in milliseconds, that the Campaign scheduler thread waits before polling the IBM Marketing Software Scheduler for jobs when IBM Campaign first starts up. Set this value to be at least as long as it takes for IBM Campaign to fully start up on your system. The default value is five minutes.

Default value

`300000`

Valid Values

Any integer

Maximum unknown status polling count

Description

Specifies the number of times the scheduler checks the status of a scheduled run whose status cannot be determined. After this limit is reached, the run status is listed as Unknown on the **Settings > Schedule management** page.

Default value

`5`

Valid Values

Any integer

Enable Scheduler

Description

Specifies whether the scheduler is enabled. Set this property to False if you do not want users to be able to use the scheduler. The False setting turns off the scheduler for all products that use it.

You must restart the Marketing Platform web application when you enable or disable the scheduler.

Default value

`True`

Valid Values

`True | False`

IBM Marketing Platform | Scheduler | Recurrence definitions

Properties in this category set the recurrence patterns for the IBM Marketing Software Scheduler. These appear in the dialog box you use if you set a recurrence pattern when you create a schedule. You can use the Recurrence template to create your own recurrence pattern, using any valid Cron expression.

Every hour

Description

The job is triggered every hour.

Default value

`0 0 0/1 * * ?`

Every day

Description

The job is triggered every 24 hours.

Default value

`0 0 0 * * ?`

Every [day of week] at 12:00 am

Description

The job is triggered on the specified day of the week at 12:00 am.

Default value

- Monday - `0 0 0 ? * MON`
- Tuesday - `0 0 0 ? * TUE`
- Wednesday - `0 0 0 ? * WED`
- Thursday - `0 0 0 ? * THU`
- Friday - `0 0 0 ? * FRI`
- Saturday - `0 0 0 ? * SAT`
- Sunday - `0 0 0 ? * SUN`

[First|Last] day of every month at 12:00 am

Description

The job is triggered on the specified day of the month (first or last) at 12:00 am.

Default value

- First day of every month - `0 0 0 1 * ?`
- Last day of every month - `0 0 0 L * ?`

[First|Last] day of every quarter at 12:00 am

Description

The job is triggered on the specified day of the calendar quarter (first or last day) at 12:00 am.

Default value

- First day of every quarter - `0 0 0 1 * JAN,APR,JUL,OCT`
- Last day of every quarter - `0 0 0 L * MAR,JUN,SEP,DEC`

[First|Last] day of every year at 12:00 am

Description

The job is triggered on the specified day of the year (first or last) at 12:00 am.

Default value

- First day of every year - `0 0 0 1 ? JAN *`
- Last day of every year - `0 0 0 L ? DEC *`

Every [month] at 12:00 am

Description

The job is triggered on the first day of the specified month at 12:00 am.

Default value

- Every January - `0 0 0 1 ? JAN *`
- Every February - `0 0 0 1 ? FEB *`
- Every March - `0 0 0 1 ? MAR *`
- Every April - `0 0 0 1 ? APR *`
- Every May - `0 0 0 1 ? MAY *`
- Every June - `0 0 0 1 ? JUN *`
- Every July - `0 0 0 1 ? JUL *`
- Every August - `0 0 0 1 ? AUG *`
- Every September - `0 0 0 1 ? SEP *`
- Every October - `0 0 0 1 ? OCT *`
- Every November - `0 0 0 1 ? NOV *`
- Every December - `0 0 0 1 ? DEC *`

IBM Marketing Platform | Scheduler | Schedule registrations | [Product] | [Object type]

A different category exists for each of the object types that can be scheduled with the IBM Marketing Software Scheduler. Properties in these categories should not normally be changed.

Executor class name

Description

The class that the IBM Marketing Software Scheduler uses to trigger a flowchart or mailing run.

Default value

Status polling interval

Configuration category

`Platform|Scheduler|Schedule registrations|[Product]|[Object type]`

For IBM Campaign flowcharts, the path for this property is `Platform|Scheduler|Schedule registrations|Campaign|Flowchart`

Description

The IBM Marketing Software Scheduler polls the product at regular intervals to obtain the run status of scheduled objects (for example, flowcharts or mailings) that have not reported a status. The interval is specified in milliseconds. The default value is 10 minutes. A more frequent polling interval (a smaller value) can negatively affect system performance. A less frequent polling interval (a larger value) reduces the load on the system. For IBM Campaign, set a less frequent polling interval when you have a large number of Campaign flowcharts that take more than 10 minutes to complete.

Default value

`600000`

Name of group to receive job notifications

Description

Notifications for all schedules for each object type are sent to all members of the group you specify here.

IBM Marketing Platform | Scheduler | Schedule registrations | [Product] | [Object type] | [Throttling group]

Default throttling groups exist for each of the object types that can be scheduled with the IBM Marketing Software scheduler. Note that these default groups do not appear on the User groups page. You can use the throttling group template to create additional groups.

Throttling threshold

Description

The greatest number of schedules associated with this group that can run concurrently. The groups you specify here appear in the **Scheduler group** drop-down list in the scheduler user interface for creating and editing schedules. The default throttling group is set to `999`, which is effectively no

limit. Because all schedules must belong to a throttling group, you should leave this value unchanged so that schedules that you do not want to throttle can be assigned to this group.

Default value

Valid Values

Any positive integer.

IBM Marketing Platform | Security

The property in this category specifies the login mode for IBM Marketing Software products.

Login method

Description

Specifies the authentication mode for all IBM Marketing Software products installed and configured to work together, as follows:

- If you set the value to `IBM Marketing Platform`, IBM Marketing Software products use the Marketing Platform for authentication and authorization.
- If you set the value to `LDAP`, IBM Marketing Software products use an LDAP server for authentication.
- If you set the value to `Web access control`, IBM Marketing Software products use web access control software for authentication.
- If you set the value to `SAML 2.0`, IBM Marketing Software products use your IdP server for authentication.

If you change this setting, stop and restart the Marketing Platform web application so that your change takes effect.

Default value

`IBM Marketing Platform`

Valid Values

`IBM Marketing Platform | LDAP | Web access control`

IBM Marketing Platform | Security | Login method details | LDAP

Properties in this category are used to configure LDAP integration.

LDAP server host name

Description

Specifies the name or IP address of the LDAP server. Set the value to the machine name or IP address of the LDAP server. For example: `machineName.companyDomain.com`

If you are integrating with Windows Active Directory, use the server name instead of the DNS name.

Default value

Undefined

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

LDAP server port

Description

Specifies the port on which the LDAP server listens. Set the value to the appropriate port number. Typically, the port number is 389 (636 if SSL is used).

Default value

389

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

User search filter

Description

Specifies the filter to use to search for users. Valid values are any valid LDAP search filter (see RFC 2254). Note that you must XML-escape any XML characters in this value.

Typically, the value for the user login attribute is uid for LDAP servers and sAMAccountName for Windows Active Directory servers. You should verify this on your LDAP or Active Directory server. If your LDAP server is Windows Active Directory, you should change the default value of this property to use sAMAccountName rather than uid. For example:

```
(&(|(objectClass=user)(objectClass=person))(sAMAccountName={0}))
```

Default value

```
(&(|(objectClass=user)(objectClass=person))(uid={0}))
```

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Use credentials stored in IBM Marketing Platform

Description

Specifies whether the Marketing Platform uses credentials from the Marketing Platform database when searching the LDAP or Windows Active Directory server during user authentication (at login time).

If this value is true, the Marketing Platform uses credentials from the Marketing Platform database, and you must set the appropriate values for the IBM Marketing Platform user for LDAP credentials and Data source for LDAP credentials properties in this category.

If your LDAP or Windows Active Directory server does not allow anonymous access, set this value to true.

If this value is false, the Marketing Platform connects with the LDAP or Windows Active Directory server anonymously. You may set this value to false if your LDAP or Windows Active Directory server allows anonymous access.

Default value

false

Valid Values

true | false

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

IBM Marketing Platform user for LDAP credentials

Description

Specifies the name of the IBM Marketing Software user that has been given LDAP administrator login credentials. Set this value if you set the Use credentials stored in UIBM Marketing Platform property in this category to true.

Set the value of this property to the user name you created for the IBM Marketing Software user when you configured LDAP integration. This property works in conjunction with the Data source for LDAP credentials property in this category.

Default value

asm_admin

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Data source for LDAP credentials

Description

Specifies the Marketing Platform data source for LDAP administrator credentials. Set this value if you set the Use credentials stored in IBM Marketing Platform property in this category to true.

Set the value of this property to the data source name you created for the IBM Marketing Software user when you configured LDAP integration. This property works in conjunction with the IBM Marketing Platform user for LDAP credentials property in this category.

Default value

Undefined

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Base DN

Description

Specifies the base distinguishing name (DN) pointing to the root of the LDAP directory structure.

Default value

[CHANGE ME]

Valid Values

Any valid DN (see RFC 1779, RFC 2253)

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Require SSL for LDAP connection

Path

IBM Marketing Platform | Security | LDAP

Description

Specifies whether the Marketing Platform uses SSL when it connects to the LDAP server to authenticate users. If you set the value to `true` , the connection is secured using SSL.

Default value

`false`

Valid Values

`true | false`

Related tasks:

"Setting LDAP login method connection properties in IBM Marketing Software" on page 124

IBM Marketing Platform | Security | Login method details | Web access control

Properties in this category are used to configure integration with web access control software.

Username pattern

Description

Java regular expression used to extract the user login from the HTTP header variable in web access control software. Note that you must XML-escape any XML characters in the regular expression. The recommended value for SiteMinder and IBM Security Access Manager is `\w*`

You should also use this value when you use a custom proxy to integrate Campaign hosted on premises and Digital Analytics in the cloud.

Default value

Undefined

Valid Values

Any Java regular expression.

Availability

This property is used only when the Marketing Platform is configured to integrate with web access control software.

Web access control header variable

Description

Specifies the HTTP header variable configured in the web access control software, which is submitted to the web application server. By default, SiteMinder uses `sm_user` and IBM Security Access Manager (SAM) uses `iv-user`. For SAM, set this value to the user name component of the IBM Raw string, not the IBM HTTP string.

Default value

Undefined

Valid Values

Any string

Availability

This property is used only when the Marketing Platform is configured to integrate with web access control software.

Related tasks:

"Setting the security mode to enable NTLMv2 authentication" on page 127

"Setting web access control connection properties in IBM Marketing Software" on page 147

IBM Marketing Platform | Security | Login method details | SAML 2.0

Properties in this category configure single sign-on through a SAML 2.0 IdP server.

IdP server URL for single sign-in

Description

The URL of the page that appears when users open the single sign-on URL to IBM Marketing Software.

Default value

`[CHANGE ME]`

IdP server URL for single sign-out

Description

Optional. When users log out, they can be redirected to the page you set here so that their logout also logs them out of the IdP server. Your IdP server is likely to provide a URL for this purpose.

Default value

`[CHANGE ME]`

Error page URL for SSO error

Description

If an error occurs during single sign-on due to a configuration or integration issue, users can be redirected to the page specified here. This setting overrides the default error page provided by Marketing Platform.

Default value
[CHANGE ME]

Destination URL

Description
The URL of the service provider (application) to which the user is redirected after successful authentication through the IdP server. This URL appears in every SAML request under the <AuthnRequest Destination> tag.

Default value
[CHANGE ME]

Consumer service URL

Description
The assertion consumer service URL that the service provider (application) consumes and parses for SAML assertions. This URL appears in every SAML request under the <AuthnRequest AssertionConsumerServiceURL> tag. This value can be the same as the value of the **Destination URL** property.

Default value
[CHANGE ME]

Application ID

Description
The application ID assigned to Marketing Platform in the IdP server. This ID is included in every SAML request to the IdP server. This ID appears in every SAML request under the <Issuer> tag.

Default value
[CHANGE ME]

Service provider name qualifier

Description
The service provider's name qualifier. This name qualifier appears in every SAML request under the <NameIDPolicy SPNameQualifier> tag.

Default value
[CHANGE ME]

Metadata path

Description
The location of the metadata file on the Marketing Platform server.

Default value
[CHANGE ME]

Entity ID

Description
The entity ID of the IdP server. Set this property to the value of *entityID* in the XML declaration at the top of the metadata file produced by the IdP server.

Marketing Platform uses this ID during assertion validation to load the IdP configurations and digital certificate.

Default value

[CHANGE ME]

Attributes NVP for response parsing

Description

User account attributes are sent to Marketing Platform by the IdP server. You can use this configuration property to capture attributes for users created in Marketing Platform automatically, when the **Add authenticated users to Marketing Platform** property is enabled.

The IdP server might use a different name for an attribute compared to the name that Marketing Platform uses. You can use this property to map the IdP attribute to the corresponding attribute in Marketing Platform. This eliminates the need for code changes.

For example, the IdP server might use **emailAddress** as the name for an attribute that is named **Email** in Marketing Platform. You would enter **Email=emailAddress** as a value in this property to map the attribute.

Use the following values for the user attributes in Marketing Platform.

- FirstName
- LastName
- Department
- Organization
- Country
- Email
- Address1
- Address2
- Phone1

  Use for work phone.
- Phone2

  Use for mobile phone.
- Phone3

  Use for home phone.
- AltLogin
- ExternalUsersGroup

  If you enable the **Add authenticated users to Marketing Platform** property, a user authenticated from the IdP server is created in Marketing Platform if that user does not already have a Marketing Platform account. These users are automatically added to a default user group, **ExternalUsersGroup**. However, you can also specify a custom group to which users are added. If you implement this option, set the value of the **ExternalUsersGroup** attribute to the name of the custom user group. For example, if you want a user to be added to a group named MyGroup, you would set this value to `ExternalUserGroup=MyGroup`.

Separate multiple name-value pairs with a semi-colon.

Default value

```
omit-xml-declaration=yes;
```

Process encrypted IdP response

Description

If your IdP server is configured to send encrypted responses, enable this property to indicate that the SAML response from the IdP server must be decrypted using the configured shared key before Marketing Platform processes it.

If you enable this property, you must also set the value of **Shared secret key** to the secret key that is used to decrypt the response.

Default value

`Disabled`

Shared secret key

Description

When the **Process encrypted IdP response** option is enabled, set this property value to the path of the keystore file.

Default value

`[CHANGE ME]`

Key store credential holder

Description

Set this value to the login name of the IBM Marketing Software user account that holds the SAML shared secret in a data source.

Default value

`[CHANGE ME]`

Key store credential data source

Description

Set this value to the name of the data source created to hold the shared secret used for decryption. The password in the data source is the password for the key store file.

Default value

`[CHANGE ME]`

Certificate alias

Description

When the **Process encrypted IdP response** option is enabled, set this property value to the certificate alias of the private key stored in the keystore file. This is used in decrypting the encrypted SAML response sent by the IDP server.

Default value

`[CHANGE ME]`

Add authenticated users to Marketing Platform

Description

When this option is enabled, a user authenticated from the IdP server is created in Marketing Platform if that user does not already have a Marketing Platform account.

Newly created users are automatically added to a default group, **ExternalUsersGroup**.

The **ExternalUsersGroup** has only the Marketing Platform **UserRole**. An administrator must grant additional permissions for the newly created users to access and use IBM Marketing Software products. An administrator can grant additional permissions by making users members of groups with different application access levels.

Alternatively, the SAML response can contain a custom user group name, and newly created users are added to this group.

When this option is disabled, a user authenticated from the IdP server can not access Marketing Platform, if that user does not have an account in Marketing Platform.

Default value

`Disabled`

Redirect to SSO

Description

When this value is **True**:

- Users who log in to IBM Marketing Software are redirected to the IdP single sign on page
- After users log in, they go to the standard Marketing Platform landing page.
- The standard Marketing Platform login screen is never available

Related concepts:

Chapter 8, "SAML 2.0 single sign-on," on page 99

Related tasks:

"Setting SAML 2.0 configuration properties" on page 102

IBM Marketing Platform | Security | LDAP synchronization

LDAP synchronization properties specify details that the system uses to log into the directory server and identify users to import. Some of these properties also control the frequency and other details of the automatic synchronization process.

LDAP sync enabled

Description

Set to `true` to enable LDAP or Active Directory synchronization.

Default value

`false`

Valid Values

`true | false`

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

LDAP sync interval

Description

The Marketing Platform synchronizes with the LDAP or Active Directory server at regular intervals, specified in seconds here. If the value is zero or less, the Marketing Platform does not synchronize. If the value is a positive integer, the new value takes effect without a restart within ten minutes. Subsequent changes take effect within the configured interval time.

Default value

600, or ten minutes

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

LDAP sync delay

Description

This the time (in 24 hour format) after which the periodic synchronization with the LDAP server begins, after the Marketing Platform is started. For example an `LDAP sync delay` of 23:00 and an`LDAP sync interval` of 600 mean that when the Marketing Platform starts, the periodic synchronization starts to execute at 11:00 PM and executes every 10 minutes (600 seconds) thereafter.

Default value

`23:00`, or 11:00pm

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

LDAP sync timeout

Description

The LDAP sync timeout property specifies the maximum length of time, in minutes, after the start of a synchronization before the Marketing Platform marks the process ended. The Platform allows only one synchronization process to run at a time. If a synchronization fails, it is marked as ended whether it completed successfully or not.

This is most useful in a clustered environment. For example, if the Marketing Platform is deployed in a cluster, one server in the cluster might start an LDAP synchronization and then go down before the process is marked as ended. In that case, the Marketing Platform will wait for the amount of time specified in this property, and then it will start the next scheduled synchronization.

Default value

600, (600 minutes, or ten hours)

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

LDAP sync scope

Description

Controls the scope of the initial query to retrieve the set of users. You should retain the default value of SUBTREE for synchronizing with most LDAP servers.

Default value

SUBTREE

Valid Values

The values are standard LDAP search scope terms.

- OBJECT - Search only the entry at the base DN, resulting in only that entry being returned
- ONE_LEVEL - Search all entries one level under the base DN, but not including the base DN.
- SUBTREE - Search all entries at all levels under and including the specified base DN.

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

LDAP provider URL

Description

For most implementations, set to the LDAP URL of the LDAP or Active Directory server, in one of the following forms:

- ldap://IP_address:port_number
- ldap://machineName.domain.com:port_number

On LDAP servers, the port number is typically 389 (636 if SSL is used).

If IBM Marketing Software is integrated with an Active Directory server, and your Active Directory implementation uses serverless bind, set the value of this property to the URL for your Active Directory server, using the following form:

ldap:///dc=example,dc=com

Default value

Undefined

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Require SSL for LDAP connection

Path

IBM Marketing Platform | Security | LDAP synchronization

Description

Specifies whether the Marketing Platform uses SSL when it connects to the LDAP server to synchronize users. If you set the value to true, the connection is secured using SSL.

Default value

false

Valid Values

`true | false`

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

LDAP config IBM Marketing Platform group delimiter

Description

In the `LDAP reference to IBM Marketing Platform group map` category, if you want to map one LDAP or Active Directory group to multiple Marketing Platform groups, use the delimiter specified here. It can be any single character that does not appear in the names it is separating.

Default value

; (semicolon)

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

LDAP reference config delimiter

Description

Specifies the delimiter that separates the `SEARCHBASE` and `FILTER` components that make up the LDAP or Active Directory reference (described in the `LDAP references for IBM Marketing Platform user creation` category).

`FILTER` is optional: if omitted, the Marketing Platform server dynamically creates the filter based on the value of the `LDAP user reference attribute name` property.

Default value

; (semicolon)

Valid Values

Any single character that does not appear in the names it is separating.

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

IBM Marketing Platform user for LDAP credentials

Description

Specifies the name of IBM Marketing Software user that has been given LDAP administrator login credentials.

Set the value of this property to the user name you created for the IBM Marketing Software user when you configured LDAP integration. This property works in conjunction with the `Data source for LDAP credentials` property in this category.

Default value

asm_admin

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Data source for LDAP credentials

Description

Specifies the Marketing Platform data source for LDAP administrator credentials.

Set the value of this property to the data source name you created for the IBM Marketing Software user when you configured LDAP integration. This property works in conjunction with the IBM Marketing Platform user for LDAP credentials property in this category.

Default value

Undefined

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

LDAP user reference attribute name

Description

For group based import of users, set to the name that your LDAP or Active Directory server uses for the user attribute in the Group object. Typically, this value is uniquemember in LDAP servers and member in Windows Active Directory servers.

For attribute based import of users, set this property to DN, and when you configure the **LDAP reference map** property, set the FILTER portion of the value to the string your LDAP server uses for the attribute on which you want to search.

Default value

member

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

LDAP BaseDN periodic search enabled

Description

When this property is set to True, the Marketing Platform performs the LDAP synchronization search using the distinguished name set in the Base DN property under the **IBM Marketing Platform | Security | LDAP** category. If this property is set to False, the Marketing Platform performs the LDAP synchronization search using the groups mapped to LDAP groups under **LDAP reference to IBM Marketing Platform group map**.

The following table describes whether changes are picked up in periodic synchronization, depending on the value set for this property.

Table 78. Effect of this property on periodic synchronization behavior

Change	Is the change picked up when the value is set to True?	Is the change picked up when the value is set to False?
In Marketing Platform, delete a user synchronized from the LDAP server	Yes	No
Remove a user from an LDAP group mapped to a Marketing Platform group	No	No
In Marketing Platform, remove a user from a Marketing Platform group mapped to an LDAP group.	No	No
Add a new user to the LDAP server	Yes	Yes
Add a user to an LDAP group mapped to a Marketing Platform group	Yes	No
Change user attributes on the LDAP server	Yes	Yes

Default value

`True`

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

User login

Description

Maps the IBM Marketing Software user's login to the equivalent user attribute in your LDAP or Active Directory server. `User login` is the only required mapping. Typically, the value for this attribute is `uid` for LDAP servers and `sAMAccountName` for Windows Active Directory servers. You should verify this on your LDAP or Active Directory server.

Default value

`uid`

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

First name

Description

Maps the First Name user attribute in the Marketing Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

`givenName`

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Last name

Description

Maps the Last Name user attribute in the Marketing Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

`sn`

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

User title

Description

Maps the Title user attribute in the Marketing Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

`title`

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Department

Description

Maps the Department user attribute in the Marketing Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

Undefined

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Company

Description

Maps the Company user attribute in the Marketing Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

Undefined

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Country

Description

Maps the Country user attribute in the Marketing Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

Undefined

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

User email

Description

Maps the Email Address attribute in the Marketing Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

`mail`

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Address 1

Description

Maps the Address user attribute in the Marketing Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

Undefined

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Work phone

Description

Maps the Work Phone user attribute in the Marketing Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

`telephoneNumber`

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Mobile phone

Description

Maps the Mobile Phone user attribute in the Marketing Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

Undefined

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Home phone

Description

Maps the Home Phone user attribute in the Marketing Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

Undefined

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Alternate login

Description

Maps the Alternate Login user attribute in the Marketing Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

Undefined

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Related tasks:

"Setting LDAP synchronization properties" on page 124

"Setting user attributes map properties" on page 125

IBM Marketing Platform | Security | LDAP synchronization | LDAP reference to IBM Marketing Platform group map

Properties in this category are used to configure LDAP integration.

LDAP reference map

Description

Users who are members of the LDAP or Active Directory group specified here are imported to the Marketing Platform group specified in the `IBM Marketing Platform group` property.

Set the value of this property using the following syntax: `SEARCHBASE DELIMITER FILTER` where:

`SEARCHBASE` is the Distinguished Name (DN) of the object.

`DELIMITER` is the value of the `LDAP config AM group delimiter` property.

FILTER is the LDAP or Active Directory attribute filter. FILTER is optional when you use group based import: if omitted, the Marketing Platform server dynamically creates the filter based on the value of the LDAP user reference attribute name property.

If you use attribute based import, set the value of FILTER to the string your LDAP server uses for the attribute on which you want to search. Also, you must set value of the **LDAP user reference attribute name** property to DN.

Default value

Undefined

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

IBM Marketing Platform group

Description

Users who are members of the LDAP or Active Directory group specified in the LDAP reference group property are imported to the Marketing Platform group specified here.

Default value

Undefined

Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory or other LDAP server.

Related tasks:

"Mapping LDAP groups to IBM Marketing Software groups" on page 126

IBM Marketing Platform | Security | Federated authentication

The properties in this category are used in implementing SAML (Security Assertion Markup Language) 2.0 based federated authentication, which enables single sign-on among diverse applications.

Allow federated login

Description

Select the check box in this property to enable federated authentication in an integrated environment.

Default value

Disabled

Identity provider URL

Description

The URL of the identity provider server.

Certificate issuer

Description

The URL of the Certificate Authority that issued the certificate on the

identity provider server. If you generate your own certificates using the Java keytool utility, set this value to the IdP server URL.

Related tasks:

"Setting configuration properties on the Configuration page" on page 94

IBM Marketing Platform | Security | Federated authentication | partitions | partition[n]

The properties in this category are used in implementing SAML (Security Assertion Markup Language) 2.0 based federated authentication between IBM Marketing Software applications and other IBM and third party applications.

Keystore path

Description

The location of the trusted keystore file in the web application server.

Keystore passkey

Description

The passkey for the keystore in the web application server.

Keystore alias

Description

The alias for the keystore in the web application server.

Related tasks:

"Setting configuration properties on the Configuration page" on page 94

IBM Marketing Platform | Security | API management

Properties in this category configure authentication behavior that applies to all IBM Marketing Software APIs.

Enable session-based API authentication

Description

If you select the check box for this property to enable it, users who are authenticated by logging in to IBM Marketing Software are not asked to log in again when they access a secure API from an IBM Marketing Software application during the session for which they are authenticated.

For example, when this property is enabled, and an authenticated Interact user calls a Campaign API during their session, no further login is required.

Default value

Disabled

Delete security token after a single use

Description

If you select the check box for this property to enable it, the token generated for an authenticated user is destroyed the first time this token is used for accessing any secure API. This enhances security by preventing any further use of the token.

Default value

Enabled

IBM Marketing Platform | Security | API management | [Product] | (API configuration template)

Use the templates in this category to configure authentication for IBM Marketing Software APIs. You can block access, require HTTPS, or require authentication for APIs.

API URI

Description

For each product, the first part of the URI is resolved by the security framework, as follows: `http[s]://host:port/context root/api/product`

Therefore, in this field you should enter only the resource name or names of the API you want to configure. You can obtain the string you need to enter from the product's API documentation.

The value used for this property must start with a / (forward slash); otherwise the configuration is ignored by the security framework.

This property supports an exact URL match as well a pattern match for the configured APIs.

- For an exact match, the URI may end with a forward slash (/) or the resource name.
- For a pattern match, the URI must end with an asterisk (*).

If you set the value of this property to `/*` the settings you use for the other properties in the category apply to all APIs for the product.

Note: For the Marketing Platform `login` API, this configuration property is read-only.

Default value

Undefined

Block API access

Description

Select this option when you want to prevent an API from accessing a product. This option is not selected by default.

When an API is blocked, the security filter returns the HTTP status code 403 (forbidden).

Secure API access over HTTPS

Description

Select this option when you want to allow the API to access a product only over HTTPS. This option is selected by default.

When an API with this property enabled is accessed over HTTP rather than HTTPS, the security filter returns the HTTP status code 403 (forbidden).

Require authentication for API access

Description

Select this option when you want to require an API to authenticate before it can access a product. This option is selected by default.

When an API with this property enabled is accessed with invalid credentials, the security filter returns the HTTP status code 401 (unauthorized).

Note: For the Marketing Platform `login` API, this configuration property is disabled, as this API is the first to be called for API authentication.

Related concepts:

Chapter 16, "Security framework for IBM Marketing Software APIs," on page 173

IBM Marketing Platform | Security | JWT authentication

JWT authentication is used for Journey Designer+Campaign. JWT authentication allows single sign-on between applications.

Enable JWT authentication

Description

When the check box for this property is selected, JWT authentication is enabled.

This property applies only in environments where Journey Designer is integrated with Campaign.

Default value

`disabled`

JWT service URL

Description

The URL of the JWT service. This value differs, depending on whether you have applied Marketing Platform FixPack 10.0.0.1. Refer to the following examples.

- If you have **not** applied FixPack 10.0.0.1:

  `http://IP_ADDRESS/jwt/api/v1/tokens`
- If you have applied FixPack 10.0.0.1:

  `http://IP_ADDRESS/api/v1/keys`

This property applies only in environments where Journey Designer is integrated with Campaign.

JWT shared secret

Description

The shared secret key that is sent from Marketing Platform to the JWT

service for authentication. This key is shared between Marketing Platform and Journey Designer. The JWT issuer is mapped to the JWT shared secret within the JWT service.

This property applies only in environments where Journey Designer is integrated with Campaign, and where Marketing Platform is version 10.0.0.0 (that is where Marketing Platform FixPack 10.0.0.1 is **not** applied).

JWT issuer

Description

The issuer name and version that is sent from Marketing Platform to the JWT service for authentication.

This property applies only in environments where Journey Designer is integrated with Campaign.

Related tasks:

Chapter 9, "Configuring JWT authentication between applications," on page 107

IBM Marketing Platform | Notifications

Properties in this category control system behavior for notifications that IBM Marketing Software products can send to users.

How many days to retain alerts

Description

Specifies the amount of time, in days, that a system alert is retained in the system for historical purpose after the expiry date, which is provided by the application that sent the alert. Alerts older than the specified number of days are deleted from the system.

Default value

`90`

How frequently to send emails (in minutes)

Description

Specifies how many minutes the system waits before sending any new notification emails.

Default value

`30`

Maximum re-tries for sending email

Description

Specifies how many times the system attempts to send notification emails when an initial attempt to send fails.

Default value

`1`

Related tasks:

"Configuring email notifications in IBM Marketing Software" on page 152

IBM Marketing Platform | Audit Events

The property on this page determines whether audit events are tracked.

Is event auditing enabled?

Description

Specifies whether the audit events are tracked.

Default value

`False`

Valid values

`True | False`

Related tasks:

"Configuring which audit events appear in the report" on page 208

IBM Marketing Platform | Audit Events | Audit events configuration

The events you select on this page are available in the security audit reports.

Record login and logout events for all accounts

Description

Specifies whether to track the user name and the date and time for log in and log out events for all user accounts.

Record when user sessions time out for all accounts

Description

Specifies whether to track the account user name and the date and time of sessions that are automatically timed out.

Record login and logout events for members of the HighSeverityAccounts group

Description

Specifies whether to track the user name and the date and time for log in and log out events for accounts that are members of the **highSeverityAccounts** group in Marketing Platform. To enable this feature, you must set a severity level for this configuration property and add users to the highSeverityAccounts group.

Record LDAP group membership changes

Description

Specifies whether to record the addition or deletion of accounts, along with the user names and dates and times of these actions, for user accounts synchronized from an LDAP server. This property applies only when Marketing Platform is integrated with a supported LDAP server, such IBM Security Directory server or Windows Active Directory.

Record when accounts are enabled and disabled

Description

Specifies whether to record the account user name and the date and time when user accounts are enabled or disabled.

Record when account passwords change

Description

Specifies whether to record the account user name and the date and time when user passwords change.

Record when account passwords are locked

Description

Specifies whether to record the account user name and the date and time when a password is locked out due to too many incorrect login attempts.

Record when groups are created or deleted in Marketing Platform

Description

Specifies whether to record when groups are added or deleted.

Record Marketing Platform group membership changes

Description

Specifies whether to record when user accounts are added to or removed from a group.

Record Marketing Platform group permission changes

Description

Specifies whether to record changes to group permissions.

Record role creation or deletion

Description

Specifies whether to record when roles and added or deleted. Only roles that are shown on the **Settings > User roles and permissions** page are tracked.

Record role membership changes

Description

Specifies whether to record changes in role membership. Only roles that are shown on the **Settings > User roles and permissions** page are tracked.

Record role permission changes

Description

Specifies whether to record changes in role permissions. Only roles that are shown on the **Settings > User roles and permissions** page are tracked.

Record changes to properties on the configuration page

Description

Specifies whether to record changes in configuration properties on the **Settings > Configuration** page. Changes made by users on the

Configuration page, or by users running the configTool are tracked. Configuration changes made by the installers during installation or upgrade are not tracked.

Enable audit backup

Description

Specifies whether to save audit data to the USM_AUDIT_BACKUP table.

Important: Because this is a bootstrap property that is read when the Marketing Platform web application starts, you must stop and restart the Marketing Platform web application when you change this property value.

Default value

False

Valid values

True | False

Archive data after the number of days specified here

Description

Specifies the interval, in days, between audit backups. The archived data is stored in the USM_AUDIT_BACKUP table and can be included in the Audit Events report when you set a custom date range that includes data from the archive.

Important: Because this is a bootstrap property that is read when the Marketing Platform web application starts, you must stop and restart the Marketing Platform web application when you change this property value.

Keep Audit records in primary for number days specified here

Description

Specifies how many days of data to keep in the USM_AUDIT table for the Audit Events report. When the default settings for the Audit Events report are in effect, only the data in the USM_AUDIT table is shown in the report.

Important: Because this is a bootstrap property that is read when the Marketing Platform web application starts, you must stop and restart the Marketing Platform web application when you change this property value.

Archive start time

Description

Specifies the time of day when the system moves audit data into an archive. Use the 24 hour format for this value.

Important: Because this is a bootstrap property that is read when the Marketing Platform web application starts, you must stop and restart the Marketing Platform web application when you change this property value.

Name of group to receive audit backup notifications

Description

Specifies the IBM Marketing Software group whose members should receive notification of archive backups. You can specify only one group for

this property. Users in this group can manage their subscription to this notification by going to their **Settings > Users** page and clicking **Notification subscriptions**.

Related concepts:

"Archived audit events" on page 211

Related tasks:

"Configuring which audit events appear in the report" on page 208

IBM Marketing Platform | Audit Events | Audit events severity configuration

The severity level you specify for each event on this page appears in the Audit Events report. You can use the severity level to sort and filter the report data. The events are identical to those in the **IBM Marketing Platform | Audit Events | Audit events configuration** category.

Related tasks:

"Configuring which audit events appear in the report" on page 208

Digital Analytics configuration properties

This section describes the Digital Analytics configuration properties on the Configuration page.

These configuration properties are used in configuring single sign-on between Digital Analytics and IBM Marketing Software. See the *IBM Marketing Platform Administrator's Guide* for for details about this integration.

Related tasks:

"Setting up single sign-on between IBM Marketing Software and Digital Analytics using automatic user account creation" on page 110

"Setting up single sign-on between IBM Marketing Software and Digital Analytics using manual user account creation" on page 111

Digital Analytics

The property in this category is part of the configuration for enabling single sign-on between Digital Analytics and IBM Marketing Software.

Enable Coremetrics Analytics

Description

This is part of the configuration for enabling single sign-on between Digital Analytics and IBM Marketing Software.

Set to `true` as one of the steps for enabling single sign-on.

See the *IBM Marketing Platform Administrator's Guide* for for details about this integration.

Default value

`false`

Digital Analytics | Integration | partitions | partition[n]

Properties in this category are part of the configuration for enabling single sign-on between Digital Analytics and IBM Marketing Software.

Platform user for Coremetrics account

Description

Specifies the login name of the IBM Marketing Software user account that holds the Digital Analytics shared secret in a data source.

This is part of the configuration for enabling single sign-on between Digital Analytics and IBM Marketing Software. See the *IBM Marketing Platform Administrator's Guide* for for details about this integration.

Default value

`asm_admin`

Datasource for Coremetrics account

Description

Specifies the name of the data source created to hold the Digital Analytics shared secret.

This is part of the configuration for enabling single sign-on between Digital Analytics and IBM Marketing Software. See the *IBM Marketing Platform Administrator's Guide* for for details about this integration.

Default value

`CoremetricsDS`

Report configuration properties

The report configuration properties for IBM Marketing Software are at **Settings** > **Configuration** > **Reports**.

To generate reports, the IBM Marketing Software suite integrates with IBM Cognos, a business intelligence application. You use the **Integrations** > **Cognos** properties to identify your IBM Cognos system. Then, for Campaign, eMessage, and Interact, you must configure additional properties to set up and customize the reporting schemas.

Reports | Integrations | Cognos [version]

The IBM Marketing Software suite integrates with IBM Cognos to generate reports.

This page displays properties that specify URLs and other parameters that are used by the IBM system.

Integration Name

Description

Read-only. Specifies that IBM Cognos is the third-party reporting or analytical tool that is used by the IBM Marketing Software to display the reports.

Default value

`Cognos`

Vendor

Description

Read-only. Specifies that IBM Cognos is the name of the company that provides the application that is specified by the Integration Name property.

Default value

Cognos

Version

Description

Read-only. Specifies the product version of the application that is specified by the Integration Name property.

Default value

<version>

Enabled

Description

Specifies whether IBM Cognos is enabled for the suite.

Default value

False

Valid Values

True | False

Integration Class Name

Description

Read-only. Specifies the fully qualified name of the Java class that creates the integration interface that is used to connect to the application specified by the Integration Name property.

Default value

com.unica.report.integration.cognos.CognosIntegration

Domain

Description

Specifies the fully qualified company domain name in which your Cognos server is running. For example, myCompanyDomain.com.

If your company uses subdomains, the value in this field must include the appropriate subdomain as well.

Default value

[CHANGE ME]

Valid Values

A string no longer than 1024 characters.

Portal URL

Description

Specifies the URL of the IBM Cognos Connection portal. Use a fully qualified host name, including the domain name (and subdomain, if appropriate) that is specified in the **Domain** property. For example: `http://MyReportServer.MyCompanyDomain.com/cognos<version>/cgi-bin/cognos.cgi`

You can find the URL in IBM Cognos Configuration at: **Local Configuration > Environment** .

Default value

`http://[CHANGE ME]/cognos<version>/cgi-bin/cognos.cgi`

Valid Values

A well-formed URL.

Dispatch URL

Description

Specifies the URL of the IBM Cognos Content Manager. Use a fully qualified host name, including the domain name (and subdomain, if appropriate) specified in the `Domain` property. For example: `http://MyReportServer.MyCompanyDomain.com:9300/p2pd/servlet/dispatch`

You can find the URL in Cognos Configuration at: **Local Configuration > Environment** .

Default value

`http://[CHANGE ME]:9300/p2pd/servlet/dispatch`

9300 is the default port number for the Cognos Content Manager. Be sure that the port number specified matches that used in the Cognos installation.

Valid Values

A well-formed URL.

Authentication mode

Description

Specifies whether the IBM Cognos application is using the IBM Authentication Provider, which means it relies on the Marketing Platform for authentication.

Default value

`anonymous`

Valid Values

- `anonymous`: means that authentication is disabled.
- `authenticated`: means that the communications between the IBM system and the Cognos system are secured at the machine level. You configure a single system user and configure it with the appropriate access rights. By convention, this user is named "cognos_admin."
- `authenticatedPerUser`: means that the system evaluates individual user credentials.

Authentication namespace

Description

Read only. The namespace of the IBM Authentication Provider.

Default value

UNICA

Authentication user name

Description

Specifies the login name for the reporting system user. The IBM applications log in to Cognos as this user when Cognos is configured to use the Unica Authentication provider. This user also has access to IBM Marketing Software.

This setting applies only when the **Authentication mode** property is set to **authenticated** .

Default value

cognos_admin

Authentication datasource name

Description

Specifies the name of the data source for the reporting system user that holds the Cognos login credentials.

Default value

Cognos

Enable form authentication

Description

Specifies whether form-based authentication is enabled. You set this property to True when either of the following condition is true:

- When the IBM Marketing Software is not installed in the same domain as the IBMCognos applications.
- When IBM Cognosis accessed by using an IP address (within the same network domain) instead of the Fully Qualified Hostname (which is being used to access the IBM Marketing Software applications), even if both the IBM Marketing Software applications and the IBM Cognos installation are on the same machine.

However, when the value is True, the login process to Cognos Connection passes the login name and password in clear text and therefore is not secure unless IBM Cognos and the IBM Marketing Software are configured to use SSL communication.

Even with SSL configured, the user name and password appear as clear text in the HTML source code when you "view source" in a displayed report. For this reason, you must install IBM Cognos and IBM Marketing Software in the same domain.

Default value

False

Valid Values

`True | False`

Reports | Schemas | [product] | [schema name] | SQL Configuration

The SQL script creates views or tables for the reporting schemas. The **Reports | Schemas | [product] | [schema name] | SQL Configuration** property provides information about the name of the views or tables.

Table/View Name

Description

Specifies the name of the view or table that the SQL script you generate for this reporting schema creates. As a best practice, you should not change the name for any of the standard or default Table/View names. If you do, you must also change the name of the view in the Cognos model in IBM Cognos Framework Manager.

When you create a new reporting schema for a new audience level, you must specify the names of all the new reporting tables/views.

Default value

Varies by schema

Valid Values

A string with the following restrictions.

- It can be no longer than 18 characters
- It must use all uppercase letters

You must use the following naming convention:

- Start the name with the letter "UAR"
- Add a one-letter code to represent the IBM Marketing Software application. See the list of codes, below.
- Add an underscore character
- Add the table name, including a one or two letter code to indicate the audience level
- Finish with an underscore character.

The SQL generator appends a time dimension code, if appropriate. See the following list of codes:

For example: `UARC_COPERF_DY` is the name of the reporting view or table for Campaign Offer Performance by Day.

Following is the list of IBM Marketing Software application codes.

- Campaign: C
- eMessage: E
- Interact: I
- Distributed Marketing: X
- Marketing Operations: P
- Leads: L

Following is the list of the Time Dimension Codes added by the generator.

- Hour: HR

- Day: DY
- Week: WK
- Month: MO
- Quarter: QU
- Year: YR

Reports | Schemas | Campaign

The **Reports | Schemas | Campaign** property provides information about the data source that identifies the Campaign database.

Input Datasource (JNDI)

Description

Specifies the name of the JNDI data source that identifies the Campaign database, specifically, the system tables. This data source must exist if you want to use the SQL generation tool to generate scripts that create reporting tables. The SQL generation tool can generate scripts that create reporting views without this data source, but it cannot validate them.

The database type of this data source must match the database type that you select when you generate the SQL scripts for the Campaign views or reporting tables.

Default value

`campaignPartition1DS`

Reports | Schemas | Campaign | Offer Performance

The Offer Performance Schema yields contact and response history metrics for all offers and for offers by campaign. By default, the schema is configured to generate a "summary" view (or table) across all time.

Audience Key

Description

Specifies the name of the column that is the Audience Key for the audience level that is supported by this reporting schema.

Default value

`CustomerID`

Valid Values

A string value no longer than 255 characters

If the key includes more than one column, use commas between the column names. For example, `ColumnX,ColumnY`.

Contact History Table

Description

Specifies the name of the Contact History table for the audience level that is supported by this reporting schema.

Default value

`UA_ContactHistory`

Detailed Contact History Table

Description

Specifies the name of the Detailed Contact History table for the audience level that is supported by this reporting schema.

Default value

UA_DtlContactHist

Response History Table

Description

Specifies the name of the Response History table for the audience level that is supported by this reporting schema.

Default value

UA_ResponseHistory

Over Time Variations

Description

Specifies the calendar time periods that are used by the "over time" reports supported by this schema.

Default value

Day, Month

Valid Values

Day, Week, Month, Quarter, Year

Reports | Schemas | Campaign | [schema name] | Columns | [Contact Metric] and [Response metric]

Use these properties to add contact and response metrics to the Campaign Performance or Offer Performance reporting schemas.

Column Name

Configuration category

Reports | Schemas | Campaign | [schema name] | Columns | [Contact Metric]

Description

Specifies the name to use in the reporting view or table for the column that is specified in the **Input Column Name** field.

Default value

[CHANGE ME]

Valid Values

The name can be no longer than 18 characters, it must be in all uppercase letters, and it cannot have spaces.

Function

Configuration category

Reports | Schemas | Campaign | [schema name] | Columns | [Contact Metric]

Description

Specifies how the contact metric is determined or calculated.

Default value

count

Valid Values

count, count distinct, sum, min, max, average

Input Column Name

Configuration category

Reports | Schemas | Campaign | [schema name] | Columns | [Contact Metric]

Description

The name of the column that provides the contact metric you are adding to this reporting schema.

Default value

[CHANGE ME]

Valid Values

The name of the column in the Contact History and Detailed Contact History tables.

Control Treatment Flag

Configuration category

Reports | Schemas | Campaign | [schema name] | Columns | [Contact Metric]

Description

If you use the sample IBM Cognos reports or create your own custom reports that include control groups, then each contact metric must have two columns in the reporting schema. One column represents the metric for the control group and the other column represents the metric for the target group. The value in **Control Treatment Flag** specifies whether the column in the view represents the control group or the target group.

If your reports do not include control groups, you do not need the second column for the control group.

Default value

0

Valid Values

- 0: the column represents the target group
- 1: the column represents the control group

Column Name

Configuration category

Reports | Schemas | Campaign | [schema name] | Columns | [Response Metric]

Description

Specifies the name to use in the reporting view or table for the column that is specified in the **Input Column Name** field.

Default value

[CHANGE ME]

Valid Values

The name can be no longer than 18 characters, it must be in all uppercase letters, and it cannot have spaces.

Function

Configuration category

Reports | Schemas | Campaign | [schema name] | Columns | [Response Metric]

Description

Specifies how the response metric is determined or calculated.

Default value

count

Valid Values

count, count distinct, sum, min, max, average

Input Column Name

Configuration category

Reports | Schemas | Campaign | [schema name] | Columns | [Response Metric]

Description

The name of the column that provides the response metric you are adding to this reporting schema.

Default value

[CHANGE ME]

Valid Values

The name of the column in the Response History table.

Control Treatment Flag

Configuration category

Reports | Schemas | Campaign | [schema name] | Columns | [Response Metric]

Description

If you use the standard IBM Cognos reports or create your own custom reports that include control groups, then each response metric must have two columns in the reporting schema. One column represents the response

from the control group and the other column represents the response from the target group. The value in **Control Treatment Flag** specifies whether the column in the view represents the control group or the target group.

If your reports do not include control groups, you do not need the second column for the control group.

Default value

0

Valid Values

- 0: the column represents the target group
- 1: the column represents the control group

Reports | Schemas | Campaign | Campaign Performance

The Campaign Performance schema yields contact and response history metrics at the campaign, campaign-offer, and campaign-cell level.

Audience Key

Description

Specifies the name of the column that is the Audience Key for the audience level that is supported by this reporting schema.

Default value

CustomerID

Valid Values

A string value no longer than 255 characters.

If the key includes more than one column, use commas between the column names. For example, ColumnX,ColumnY.

Contact History Table

Description

Specifies the name of the Contact History table for the audience level that is supported by this reporting schema.

Default value

UA_ContactHistory

Detailed Contact History Table

Description

Specifies the name of the Detailed Contact History table for the audience level that is supported by this reporting schema.

Default value

UA_DtlContactHist

Response History Table

Description

Specifies the name of the Response History table for the audience level that is supported by this reporting schema.

Default value

UA_ResponseHistory

Over Time Variations

Description

Specifies the calendar time periods that are used by the "over time" reports supported by this schema.

Default value

Day, Month

Valid Values

Day, Week, Month, Quarter, Year

Reports | Schemas | Campaign | Campaign Offer Response Breakout

The Campaign Offer Response Breakout schema supports reporting on campaign-detailed responses, which are broken out by response type and by offer data. This schema template gives different response counts for each custom Response Type for campaigns and offers grouped by campaign.

This schema

Response History Table

Description

Specifies the name of the Response History table for the audience level that is supported by this reporting schema.

Default value

UA_ResponseHistory

Reports | Schemas | Campaign | Campaign Offer Response Breakout | Columns | [Response Type]

Use the **Reports | Schemas | Campaign | Campaign Offer Response Breakout | Columns | [Response Type]** property to add any custom response types that you want to include in your reports to the reporting schema.

Column Name

Description

Specifies the name to use in the reporting view or table for the column that is specified in the **Response Type Code** field.

Default value

[CHANGE ME]

Valid Values

The name can be no longer than 18 characters, it must be in all uppercase letters, and it cannot have spaces.

Response Type Code

Description

The response type code for the specified response type. This value is held in the ResponseTypeCode column in the UA_UsrResponseType table.

Default value

[CHANGE ME]

Valid Values

The example response type codes are as follows:

- EXP (explore)
- CON (consider)
- CMT (commit)
- FFL (fulfill)
- USE (use)
- USB (unsubscribe)
- UKN (unknown)

Your Campaign installation may have additional custom response type codes.

Control Treatment Flag

Description

If you use the standard IBM Cognos reports provided in the IBM Marketing Software Reports Pack or custom reports that include control groups, then each response type must have two columns in the reporting schema. One column represents the response type from the control group and the other column represents the response type from the target group. The value in **Control Treatment Flag** specifies whether the column in the view represents the control group or the target group.

If your reports do not include control groups, you do not need the second column for the control group.

Default value

0

Valid Values

- 0: the column represents the target group
- 1: the column represents the control group

Reports | Schemas | Campaign | [schema name] | Columns | [Response Metric]

Use the **Reports | Schemas | Campaign | [schema name] | Columns | [Response Metric]** property to add the response metrics that you want to include in your reports to the Campaign Performance or Offer Performance reporting schemas.

Column Name

Description

Specifies the name to use in the reporting view or table for the column that is specified in the **Input Column Name** field.

Default value

`[CHANGE ME]`

Valid Values

The name can be no longer than 18 characters, it must be in all uppercase letters, and it cannot have spaces.

Function

Description

Specifies how the response metric is determined or calculated.

Default value

`count`

Valid Values

`count`, `count distinct`, `sum`, `min`, `max`, `average`

Input Column Name

Description

The name of the column that provides the response metric you are adding to this reporting schema.

Default value

`[CHANGE ME]`

Valid Values

The name of the column in the Response History table.

Control Treatment Flag

Description

If you use the standard IBM Cognos reports or create your own custom reports that include control groups, then each response metric must have two columns in the reporting schema. One column represents the response from the control group and the other column represents the response from the target group. The value in **Control Treatment Flag** specifies whether the column in the view represents the control group or the target group.

If your reports do not include control groups, you do not need the second column for the control group.

Default value

`0`

Valid Values

- `0`: the column represents the target group
- `1`: the column represents the control group

Reports | Schemas | Campaign | Campaign Offer Contact Status Breakout

The Campaign Offer Contact Status Breakout schema supports reporting on campaign-detailed contacts, which are broken out by contact status type and by offer data. This schema template gives different contact counts for each custom Contact Status Type for campaigns and offers grouped by campaign.

By default, none of the example Campaign reports use this schema.

Audience Key

Description

Specifies the name of the column that is the Audience Key for the audience level that is supported by this reporting schema.

Default value

`CustomerID`

Valid Values

A string value no longer than 255 characters.

If the key includes more than one column, use commas between the column names. For example, `ColumnX,ColumnY`.

Contact History Table

Description

Specifies the name of the Contact History table for the audience level that is supported by this reporting schema.

Default value

`UA_ContactHistory`

Detailed Contact History Table

Description

Specifies the name of the Detailed Contact History table for the audience level that is supported by this reporting schema.

Default value

`UA_DtlContactHist`

Reports | Schemas | Campaign | Campaign Offer Contact Status Breakout | Columns | [Contact Status]

Use the **Reports | Schemas | Campaign | Campaign Offer Contact Status Breakout | Columns | [Contact Status]** to add the contact status that you want to include in your reports to the reporting schemas.

Column Name

Description

Specifies the name to use in the reporting view or table for the column that is specified in the **Contact Status** field.

Default value

[CHANGE ME]

Valid Values

The name can be no longer than 18 characters, it must be in all uppercase letters, and it cannot have spaces.

Contact Status Code

Description

The name of the contact status code. This value is held in the ContactStatusCode column in the UA_ContactStatus table.

Default value

[CHANGE ME]

Valid Values

The example contact status types are as follows.

- CSD (campaign send)
- DLV (delivered)
- UNDLV (undelivered)
- CTR (control)

Your Campaign installation may have additional custom contact status types.

Reports | Schemas | Campaign | Campaign Custom Attributes | Columns

Use these properties to add any custom offer attributes that you want to include in your reports to the reporting schema.

Column Name

Configuration category

Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Campaign Custom Column]

Description

Specifies the name to use in the reporting view or table for the attribute that is identified in the **Attribute ID** field.

Default value

[CHANGE ME]

Valid Values

The name can be no longer than 18 characters, it must be in all uppercase letters, and it cannot have spaces.

Attribute ID

Configuration category

Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Campaign Custom Column]

Description

The value from the attribute's AttributeID column in the **UA_CampAttribute** table.

Default value

0

Value Type

Configuration category

Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Campaign Custom Column]

Description

The data type of the campaign attribute.

Default value

StringValue

Valid Values

StringValue, NumberValue, DatetimeValue

If this campaign attribute holds a currency value, select NumberValue.

If this campaign attribute's **Form Element Type** was set to Select Box - String in Campaign, select StringValue.

Column Name

Configuration category

Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Cell Custom Column]

Description

Specifies the name to use in the reporting view or table for the attribute that is identified in the **Attribute ID** field.

Default value

[CHANGE ME]

Valid Values

The name can be no longer than 18 characters, it must be in all uppercase letters, and it cannot have spaces.

Attribute ID

Configuration category

Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Cell Custom Column]

Description

The value from the attribute's AttributeID column in the **UA_CellAttribute** table.

Default value

0

Value Type

Configuration category

Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Cell Custom Column]

Description

The data type of the cell attribute.

Default value

StringValue

Valid Values

StringValue, NumberValue, DatetimeValue

Column Name

Configuration category

Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Offer Custom Column]

Description

Specifies the name to use in the reporting view or table for the attribute that is identified in the **Attribute ID** field.

Default value

[CHANGE ME]

Valid Values

The name can be no longer than 18 characters, it must be in all uppercase letters, and it cannot have spaces.

Attribute ID

Configuration category

Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Offer Custom Column]

Description

The value from the attribute's AttributeID column in the **UA_OfferAttribute** table.

Default value

0

Value Type

Configuration category

Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Offer Custom Column]

Description

The data type of the offer attribute.

Default value

StringValue

Valid Values

StringValue, NumberValue, DatetimeValue

If this offer attribute holds a currency value, select NumberValue.

If this offer attribute's **Form Element Type** was set to Select Box - String in Campaign, select StringValue.

Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Campaign Custom Column]

Use the **Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Campaign Custom Column]** property to add any custom campaign attributes that you want to include in your reports to the reporting schema.

Column Name

Description

Specifies the name to use in the reporting view or table for the attribute that is identified in the **Attribute ID** field.

Default value

`[CHANGE ME]`

Valid Values

The name can be no longer than 18 characters, it must be in all uppercase letters, and it cannot have spaces.

Attribute ID

Description

The value from the attribute's `AttributeID` column in the **UA_CampAttribute** table.

Default value

0

Value Type

Description

The data type of the campaign attribute.

Default value

`StringValue`

Valid Values

`StringValue`, `NumberValue`, `DatetimeValue`

If this campaign attribute holds a currency value, select `NumberValue`.

If this campaign attribute's **Form Element Type** was set to `Select Box - String` in Campaign, select `StringValue`.

Reports | Schemas | Campaign | [schema name] | Columns | [Contact Metric]

Use the **Reports | Schemas | Campaign | [schema name] | Columns | [Contact Metric]** property to add contact metrics to the Campaign Performance or Offer Performance reporting schemas.

Column Name

Description

Specifies the name to use in the reporting view or table for the column that is specified in the **Input Column Name** field.

Default value

[CHANGE ME]

Valid Values

The name can be no longer than 18 characters, it must be in all uppercase letters, and it cannot have spaces.

Function

Description

Specifies how the contact metric is determined or calculated.

Default value

count

Valid Values

count, count distinct, sum, min, max, average

Input Column Name

Description

The name of the column that provides the contact metric you are adding to this reporting schema.

Default value

[CHANGE ME]

Valid Values

The name of the column in the Contact History and Detailed Contact History tables.

Control Treatment Flag

Description

If you use the sample IBM Cognos reports or create your own custom reports that include control groups, then each contact metric must have two columns in the reporting schema. One column represents the metric for the control group and the other column represents the metric for the target group. The value in **Control Treatment Flag** specifies whether the column in the view represents the control group or the target group.

If your reports do not include control groups, you do not need the second column for the control group.

Default value

0

Valid Values

- 0: the column represents the target group
- 1: the column represents the control group

Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Offer Custom Column]

Use the **Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Offer Custom Column]** property to add any custom offer attributes that you want to include in your reports to the reporting schema.

Use this form to add

Column Name

Description

Specifies the name to use in the reporting view or table for the attribute that is identified in the **Attribute ID** field.

Default value

`[CHANGE ME]`

Valid Values

The name can be no longer than 18 characters, it must be in all uppercase letters, and it cannot have spaces.

Attribute ID

Description

The value from the attribute's `AttributeID` column in the **UA_OfferAttribute** table.

Default value

0

Value Type

Description

The data type of the offer attribute.

Default value

`StringValue`

Valid Values

`StringValue`, `NumberValue`, `DatetimeValue`

If this offer attribute holds a currency value, select `NumberValue`.

If this offer attribute's **Form Element Type** was set to `Select Box - String` in Campaign, select `StringValue`.

Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Cell Custom Column]

Use the **Reports | Schemas | Campaign | Campaign Custom Attributes | Columns | [Cell Custom Column]** property to add any custom cell attributes that you want to include in your reports to the reporting schema.

Column Name

Description

Specifies the name to use in the reporting view or table for the attribute that is identified in the **Attribute ID** field.

Default value

`[CHANGE ME]`

Valid Values

The name can be no longer than 18 characters, it must be in all uppercase letters, and it cannot have spaces.

Attribute ID

Description

The value from the attribute's `AttributeID` column in the **UA_CellAttribute** table.

Default value

0

Value Type

Description

The data type of the cell attribute.

Default value

`StringValue`

Valid Values

`StringValue`, `NumberValue`, `DatetimeValue`

Reports | Schemas | Interact

The Interact reporting schemas reference three separate databases: the design time, runtime, and learning databases. Use the **Reports | Schemas | Interact** property to specify the JNDI names of the data sources for those databases.

The data sources that are specified on this page must exist if you want to use the Reporting SQL generation tool to generate scripts that create reporting tables. The SQL generation tool can generate scripts that create reporting views without these data sources, but it cannot validate the scripts.

The database type of the data sources must match the database type that you select when you generate the SQL scripts for the views or reporting tables.

Interact Design Datasource (JNDI)

Description

Specifies the name of the JNDI data source that identifies the Interact design time database, which is also the Campaign system tables.

Default value

`campaignPartition1DS`

Interact Runtime Datasource (JNDI)

Description

Specifies the name of the JNDI data source that identifies the Interact runtime database.

Default value

`InteractRTDS`

Interact Learning Datasource (JNDI)

Description

Specifies the name of the JNDI data source that identifies the Interact learning database.

Default value

`InteractLearningDS`

Reports | Schemas | Interact | Interact Performance

The Interact Performance schema yields contact and response history metrics at the channel, channel-offer, channel-segment, channel-interaction point, interactive cell, interactive cell-offer, interactive cell-interaction point, interactive offer, interactive offer-cell, and interactive offer-interaction point levels.

Audience Key

Description

Specifies the name of the column that is the Audience Key for the audience level that is supported by this reporting schema.

Default value

`CustomerID`

Valid Values

A string value no longer than 255 characters.

If the key includes more than one column, use commas between the column names. For example, `ColumnX,ColumnY`.

Detailed Contact History Table

Description

Specifies the name of the Detailed Contact History table for the audience level that is supported by this reporting schema.

Default value

`UA_DtlContactHist`

Response History Table

Description

Specifies the name of the Response History table for the audience level that is supported by this reporting schema.

Default value

`UA_ResponseHistory`

Over Time Variations

Description

Specifies the calendar time periods that are used by the "over time" reports supported by this schema.

Default value

`Hour, Day`

Valid Values

`Hour, Day, Week, Month, Quarter, Year`

Reports | Schemas | eMessage

The **Reports | Schemas | eMessage** property specifies the name of the data source that identifies the eMessage tracking tables, which are in the Campaign system tables.

eMessage Tracking Datasource (JNDI)

Description

Specifies the name of the JNDI data source that identifies the eMessage tracking tables, which are in the Campaign system tables. This data source must exist if you want to use the Reports SQL generation tool to validate scripts that create reporting tables. The SQL generation tool can generate scripts that create reporting views without this data source, but it cannot validate them.

The database type of this data source must match the database type that you select when you generate the SQL scripts for the views or reporting tables.

Default value

`campaignPartition1DS`

IBM Marketing Operations configuration properties

This section describes the IBM Marketing Operations configuration properties on the **Settings > Configuration** page.

Marketing Operations

Properties in this category specify the default and supported locales for your installation of IBM Marketing Operations.

supportedLocales

Description

Specifies the locales available in your installation of IBM Marketing Operations. List only the locales that you are using. Each locale you list uses memory on the server. The amount of memory that is used depends on the size and number of templates.

If you add locales after the initial installation or upgrade, you must run the upgrade servlets again. See upgrade documentation for details.

If you change this value, you must stop and restart your Marketing Operations deployment before the change takes effect.

Default value

en_US

defaultLocale

Description

Specifies the supported locale in which you want IBM Marketing Operations to display for all users, unless explicitly overridden for specific users by Marketing Operations administrators.

If you change this value, you must stop and restart your Marketing Operations deployment before the change takes effect.

Default value

en_US

Marketing Operations | navigation

The properties in this category specify options for navigation, such as Uniform Resource Identifiers, URLs, and ports.

welcomePageURI

Description

The Uniform Resource Identifier of the IBM Marketing Operations index page. This value is used internally by IBM Marketing Software applications. Changes to this value are not recommended.

Default value

affiniumPlan.jsp?cat=projectlist

projectDetailpageURI

Description

The Uniform Resource Identifier of the IBM Marketing Operations detail page. This value is used internally by IBM Marketing Software applications. Changes to this value are not recommended.

Default value

blank

seedName

Description

Used internally by IBM Marketing Software applications. Changes to this value are not recommended.

Default value

Plan

type

Description

Used internally by IBM Marketing Software applications. Changes to this value are not recommended.

Default value

Plan

httpPort

Description

The port number that is used by the application server for connections to the IBM Marketing Operations application.

Default value

7001

httpsPort

Description

The port number that is used by the application server for secure connections to the IBM Marketing Operations application.

Default value

7001

serverURL

Description

The URL of the IBM Marketing Operations installation. Accepts locators with either the HTTP or HTTPS protocol.

If users access Marketing Operations with the Chrome browser, use the fully qualified domain name (FQDN) in the URL. If the FQDN is not used, the Chrome browser cannot access the product URLs.

Default value

http://<*server*>:<*port*>/plan

Note: <server> must be lowercase.

logoutURL

Description

Used internally. Changes to this value are not recommended.

IBM Marketing Platform uses this value to call the logout handler of each registered application if the user clicks the logout link in suite.

Default value

/uapsysservlet?cat=sysmodules&func=logout

displayName

Description

Used internally.

Default value

Marketing Operations

Marketing Operations | about

The configuration properties in this section list information about your IBM Marketing Operations installation. You cannot edit these properties.

displayName

Description

The display name of the product.

Value

IBM Marketing Operations

releaseNumber

Description

The currently installed release.

Value

<version>.<release>.<modification>

copyright

Description

The copyright year.

Value

<year>

os

Description

The operating system on which IBM Marketing Operations is installed.

Value *<operating system and version>*

java

Description

The current version of Java.

Value *<version>*

support

Description

Read documentation and place service requests.

Value

`http://www-947.ibm.com/support/entry/portal/open_service_request`

appServer

Description

The address of the application server on which IBM Marketing Operations is installed.

Value

<IP address>

otherString

Description

Value

blank

Marketing Operations | umoConfiguration

These properties specify information about the basic configuration of IBM Marketing Operations.

serverType

Description

Application Server Type. Used for Calendar export.

Valid values

`WEBLOGIC` or `WEBSPHERE`

Default value

<server type>

usermanagerSyncTime

Description

Time in milliseconds to between scheduled synchronizations with IBM Marketing Platform.

Default value

`10800000` (milliseconds: 3 hours)

firstMonthInFiscalYear

Description

Set to the month that you would like your account fiscal year to begin. The Summary tab for the account contains a view-only table, which lists budget information by month for the fiscal years of the account. The first month in this table is determined by this parameter.

January is represented by 0. To have your fiscal year to begin in April, set **`firstMonthInFiscalYear`** to 3.

Valid values

Integers `0` to `11`

Default value

`0`

maximumItemsToBeRetainedInRecentVisits

Description

The maximum number of links to recently viewed pages to display on the **Recent** menu.

Default value

10 (links)

maxLimitForTitleString

Description

The maximum number of characters that can display in a page title. If titles are longer than the specified number, IBM Marketing Operations clips them.

Default value

40 (characters)

maximumLimitForBulkUploadItems

Description

The maximum number of attachments you can upload at the same time.

Default value

5 (attachments)

workingDaysCalculation

Description

Controls how IBM Marketing Operations calculates durations.

Valid values

- bus: Business days only, includes working days only. Does not include weekends and days off.
- wkd: Business days + Weekends, includes working days and weekends. Does not include days off.
- off: Business days + Days off, includes all working days and days off. Does not include weekends.
- all: includes all days in the calendar.

Default value

all

validateAllWizardSteps

Description

When users create a program, project, or request with the wizard, IBM Marketing Operations automatically validates that the required fields on the current page have values. This parameter controls whether Marketing Operations validates the required fields on all pages (tabs) when a user clicks **Finish**.

Valid values

- True: Marketing Operations checks the required fields on pages that the user did not view (except workflow, tracking, and attachments). If a required field is blank, the wizard opens that page and displays an error message.
- False: Marketing Operations does not validate required fields on pages the user did not view.

Default value

`True`

enableRevisionHistoryPrompt

Description

Ensures that users are prompted to add change comments when they save a project, request, or approval.

Valid values

`True | False`

Default value

`False`

useForecastDatesInTaskCalendar

Description

Specifies the type of dates that are used when tasks display in calendar view.

Valid values

- `True`: uses forecast and actual dates to display tasks.
- `False`: uses target dates to display tasks.

Default value

`False`

copyRequestProjectCode

Description

Controls whether you want to carry the Project Code (PID) over from a request to a project. If you set this parameter to `False`, the project and the request use different codes.

Valid values

`True | False`

Default value

`True`

projectTemplateMonthlyView

Description

Controls whether the monthly view is allowed in the workflow for a project template.

Valid values

`True | False`

Default value

`False`

disableAssignmentForUnassignedReviewers

Description

Specifies how work is assigned by role for approvals. The **disableAssignmentForUnassignedReviewers** parameter controls the behavior of **Assign work by Role** on the People tab for assignment of approvers in workflow approvals.

Valid values

- True: unassigned reviewers in the People tab are not added to the approval as new steps.
  - Append option: The existing, owner-assigned approvers without an assigned role do not change. New approver steps are not added even if the People tab has reviewers with the role "unassigned."
  - Replace option: The existing owner assigned approvers without a role are replaced with a blank. New approver steps would not be added even if the people tab has reviewers with the role "unassigned."
- False: unassigned reviewers are added to the approval.
  - Append option: All reviewers without a role are appended to the approval as reviewers if the approval has owner assigned steps without defined roles.
  - Replace Option: The existing approvers of approvals are replaced with the unassigned approvers in the People tab.

Default value

False

enableApplicationLevelCaching

Description

Indicates whether application-level caching is enabled or not. For best results in a clustered environment on which multicasting of caching messages is not enabled, consider turning off application level caching for Marketing Operations.

Valid values

True | False

Default value

True

customAccessLevelEnabled

Description

Determines whether you use custom access levels (project roles) in IBM Marketing Operations.

Valid values

- True: user access to projects and requests is evaluated according to Object Access Levels and Custom Access Levels (project roles). Tab security is enabled for custom tabs.
- False: user access to projects and requests is evaluated according only to Object Access Levels (object implicit roles), and tab security is turned off for custom tabs.

Default value

True

enableUniqueIdsAcrossTemplatizableObjects

Description

Determines whether you use unique internal IDs for all of the objects that are created from templates, including programs, projects, plans, and invoices.

Valid values

- `True` enables unique internal IDs across all objects that are created from templates. This configuration simplifies cross-object reporting by allowing the system to use the same table for different object types.
- `False` disables unique internal IDs across all objects that are created from templates.

Default value

`True`

FMEnabled

Description

Enables and disables the Financial Management Module, which determines whether the Accounts, Invoices, and Budget tabs appear in the product.

Valid values

`True | False`

Default value

`False`

FMProjVendorEnabled

Description

Parameter that is used to show/hide vendor column for project line items.

Valid values

`True | False`

Default value

`False`

FMPrgmVendorEnabled

Description

Parameter that is used to show/hide vendor column for program line items.

Valid values

`True | False`

Default value

`False`

Marketing Operations | umoConfiguration | Approvals

These properties specify options for approvals.

specifyDenyReasons

Description

Enables a customizable list of reasons for denying an approval. When enabled, administrators populate the Approval Deny Reasons list with options, then associate deny reasons with each workflow template and each project template that defines a workflow. Users who deny an approval, or an item within an approval, are required to select one of these predefined reasons.

Valid values

`True | False`

Default value

`False`

10.0.0.2

approveWithChanges

Description

Enables the **Approve with changes** option for an approval. When enabled, The **Allow approvers to approve with changes** option is selected by default when a user sets up approvals in a project template, project, or a standalone approval. The **Allow approvers to approve with changes** option can be edited if the **overrideApproveWithChanges** property is set to `True`.

An approver can approve a task by selecting the **Approve with changes** option if the **Allow approvers to approve with changes** option is selected when the approval is set up.

Valid values

`True | False`

Default value

`True`

10.0.0.2

overrideApproveWithChanges

Description

Set to `True` to allow a user to edit the default setting for the **Allow approvers to approve with changes** option when a user sets up approvals in a project template, project, or a standalone approval. The default setting is determined by the **approveWithChanges** property.

Valid values

`True | False`

Default value

`True`

Marketing Operations | umoConfiguration | templates

These properties specify information about templates in IBM Marketing Operations. For best results, do not change the default values of these parameters.

templatesDir

Description

Identifies the directory that contains all of your project template definitions, which are stored in XML files.

Use a fully qualified path.

Default value

<IBM_IMS_Home>/*<MarketingOperations_Home>*/templates

assetTemplatesFile

Description

The XML file that defines the templates for assets. This file must be in the directory that is specified by **templatesDir**.

Default value

asset_templates.xml

planTemplatesFile

Description

The XML file that defines the templates for plans. This file must be in the directory that is specified by **templatesDir**.

Default value

plan_templates.xml

programTemplatesFile

Description

The XML file that defines the templates for programs. This file must be in the directory that is specified by **templatesDir**.

Default value

program_templates.xml

projectTemplatesFile

Description

The XML file that defines the templates for projects. This file must be in the directory that is specified by **templatesDir**.

Default value

project_templates.xml

invoiceTemplatesFile

Description

The XML file that defines the templates for invoices. This file must be in the directory that is specified by **templatesDir**.

Default value

invoice_templates.xml

componentTemplatesFile

Description

The XML file that defines the templates for custom marketing object types. This file must be in the directory that is specified by **templatesDir**.

Default value

component_templates.xml

metricsTemplateFile

Description

The XML file that defines the templates for metrics. This file must be in the directory that is specified by **templatesDir**.

Default value

metric_definition.xml

teamTemplatesFile

Description

The XML file that defines the templates for teams. This file must be in the directory that is specified by **templatesDir**.

Default value

team_templates.xml

offerTemplatesFile

Description

The XML file that defines the templates for offers. This file must be in the directory that is specified by **templatesDir**.

Default value

uap_sys_default_offer_comp_type_templates.xml

Marketing Operations | umoConfiguration | attachmentFolders

These properties specify directories to upload and store attachments.

uploadDir

Description

The upload directory where attachments for projects are stored.

Default value

<MarketingOperations_Home>/projectattachments

planUploadDir

Description

The upload directory where attachments for plans are stored.

Default value

<MarketingOperations_Home>/planattachments

programUploadDir

Description

The upload directory where attachments for programs are stored.

Default value

<MarketingOperations_Home>/programattachments

componentUploadDir

Description

The upload directory where attachments for marketing objects are stored.

Default value

<MarketingOperations_Home>/componentattachments

taskUploadDir

Description

The upload directory where attachments for tasks are stored.

Default value

<MarketingOperations_Home>/taskattachments

approvalUploadDir

Description

The upload directory where approval items are stored.

Default value

<MarketingOperations_Home>/approvalitems

assetUploadDir

Description

The upload directory where assets are stored.

Default value

<MarketingOperations_Home>/assets

accountUploadDir

Description

The upload directory where attachments for accounts are stored.

Default value

<MarketingOperations_Home>/accountattachments

invoiceUploadDir

Description

The upload directory where attachments for invoices are stored.

Default value

<MarketingOperations_Home>/invoiceattachments

graphicalRefUploadDir

Description

The upload directory where attribute images are stored.

Default value

<MarketingOperations_Home>/graphicalrefimages

templateImageDir

Description

The upload directory where template images are stored.

Default value

<MarketingOperations_Home>/images

recentDataDir

Description

The temporary directory that stores the recent data (serialized) for each user.

Default value

<MarketingOperations_Home>/recentdata

workingAreaDir

Description

The temporary directory that stores CSV files that are uploaded during grid imports.

Default value

<MarketingOperations_Home>/umotemp

managedListDir

Description

The upload directory where managed list definitions are stored.

Default value

<MarketingOperations_Home>/managedList

Marketing Operations | umoConfiguration | fileUpload

The properties in this category specify options for file uploads.

validateFileUpload

Description

You can choose `True` to validate a file that is uploaded or `False` if you do not want to validate a file that is uploaded.

allowedFileTypes

Description

The type of file allowed to be uploaded. File types can include `.doc`, `.ppt`, `.xls`, `.pdf`, `.gif`, `.jpeg`, `.png`, and `.mpp`.

fileMaxSize

Description

The maximum size allowed for the file that you upload.

Marketing Operations | umoConfiguration | Email

These properties specify information for sending email notifications in IBM Marketing Operations.

notifyEMailMonitorJavaMailHost

Description

Optional string that specifies either the DNS host name of the email notifications mail server or its dot-formatted IP address. Set to the machine name or IP address of your SMTP server.

This parameter is necessary if you did not provide IBM Marketing Operations with an existing JavaMail session that uses the session parameter and the delegate is marked "Complete."

Default value

`[CHANGE-ME]`

notifyDefaultSenderEmailAddress

Description

Set to a valid email address. The system sends email messages to this address when there is no valid email address available to send the notification email messages.

Default value

`[CHANGE-ME]`

notifySenderAddressOverride

Description

Use this parameter to specify a standard value for the REPLY-TO and FROM email addresses for notifications. By default, these addresses are populated with the email address of the event owner.

Default value

blank

Marketing Operations | umoConfiguration | markup

These properties specify markup options. IBM Marketing Operations provides markup tools for making comments on attachments. You can use either Adobe Acrobat markup or native Marketing Operations markup. Use the properties in this category to configure which option to use.

markupServerType

Description

Determines which markup option to use.

Valid values

- SOAP enables users to edit and view markups in PDF documents. Adobe Acrobat Professional is required for markups. If specified, users cannot view markups that were made previously in a web browser with the native Marketing Operations method.

  If you specify SOAP, you must also configure the **markupServerURL** parameter.

  If you specify SOAP, you must delete the customized UMO_Markup_Collaboration.js that was copied in the JavaScripts subdirectory of the directory where Adobe Acrobat is installed. For example: C:\Program files (x86)\Adobe\Acrobat 10.0\Acrobat\Javascripts\UMO_Markup_Collaboration.js. This file is no longer required.
- MCM enables the native Marketing Operations markup method that allows users to edit and view markups in a web browser. If specified, users cannot edit or view markups that were made previously in a PDF with Adobe Acrobat.
- If blank, the markup function is disabled and the **View/Add Markup** link does not appear.

Default value

MCM

markupServerURL

Description

Dependent on **markupServerType** = SOAP.

Set to the URL for the computer that hosts the markup server, including the number of the port the web application server uses for listening. The URL must contain the fully qualified host name.

Accepts locators with either the HTTP or HTTPS protocol.

Default value

http://<*server*>:<*port*>/plan/services/collabService?wsdl

instantMarkupFileConversion

Description

If True, IBM Marketing Operations converts PDF attachments to images as soon as they are uploaded, rather than doing this conversion the first time a user opens the item for markup.

Valid values

True | False

Default value

False

Marketing Operations | umoConfiguration | grid

These properties specify options for grids.

gridmaxrow

Description

An optional integer to define the maximum number of rows to be retrieved in grids. The default, -1, retrieves all rows.

Default value

-1

reloadRuleFile

Description

An optional boolean parameter that indicates whether the grid validation plug-in needs to be reloaded or not.

Valid values

True | False

Default value

True

gridDataValidationClass

Description

An optional parameter to specify custom grid data validation class. If not specified, the default, the built in plug-in is used for grid data validation.

Default value

blank

tvcDataImportFieldDelimiterCSV

Description

Delimiter to use to parse data imported into a grid. Default is comma (,).

Default value

, (comma)

maximumFileSizeToImportCSVFile

Description

Represents the maximum file size in MB that can be uploaded while importing comma-separated data for TVC.

Default value

0 (unlimited)

maximumRowsToBeDisplayedPerPageInGridView

Description

Specifies the number of rows to display per page in grid view.

Valid values

positive integers

Default value

100

griddataxsd

Description

Name of grid data XSD file.

Default value

griddataschema.xsd

gridpluginxsd

Description

Name of grid plug-ins XSD file.

Default value

gridplugin.xsd

gridrulesxsd

Description

Name of grid rules XSD file.

Default value

gridrules.xsd

Marketing Operations | umoConfiguration | workflow

These properties specify options for the workflow in IBM Marketing Operations.

hideDetailedDateTime

Description

Optional show/hide parameter for detailed date time in the tasks page.

Valid values

True | False

Default value

False

daysInPastRecentTask

Description

This parameter determines how long tasks are considered "recent." If the task is "active" and started less than this number of days ago, or the Target End Date of the task is between today and this number of days in the past, the task displays as a recent task.

Valid values

positive integers

Default value

14 (days)

daysInFutureUpcomingTasks

Description

This parameter determines how many days in the future to look for upcoming tasks. If the task starts in the next **daysInFutureUpcomingTasks** or does not end before the current date, it is an upcoming task.

Valid values

positive integers

Default value

14 (days)

beginningOfDay

Description

Begin hour of the working day. This parameter is used to calculate the datetimes in workflow using fractional durations.

Valid values

integers from 0 to 12

Default value

9 (9 AM)

numberOfHoursPerDay

Description

Number of hours per day. This parameter is used to calculate the datetimes in workflow using fractional durations.

Valid values

integers from 1 to 24

Default value

8 (hours)

mileStoneRowBGColor

Description

Defines the background color for workflow tasks. To specify the value, insert the # character before the six-character Hex code for the color. For example, #0099CC.

Default value

#DDDDDD

Marketing Operations | umoConfiguration | integrationServices

These properties specify information about IBM Marketing Operations Integration Services module. The Integration Services module extends the function of Marketing Operations with web services and triggers.

enableIntegrationServices

Description

Enables and disables the Integration Services module.

Valid values

`True | False`

Default value

`False`

integrationProcedureDefinitionPath

Description

Optional full file path to the custom procedure definition XML file.

Default value

[plan-home]`/devkits/integration/examples/src/procedure/procedure-plugins.xml`

integrationProcedureClasspathURL

Description

URL to the class path for custom procedures.

Default value

`file:///`*[plan-home]*`/devkits/integration/examples/classes/`

Marketing Operations | umoConfiguration | campaignIntegration

The properties in this category specify options for the Campaign integration.

defaultCampaignPartition

Description

When IBM Marketing Operations is integrated with IBM Campaign, this parameter specifies the default Campaign partition if the campaign-partition-id is not defined in the project template.

Default value

`partition1`

webServiceTimeoutInMilliseconds

Description

Added for Web Service integration API calls. This parameter is used as a timeout for web services API calls.

Default value

`1800000` milliseconds (30 minutes)

Marketing Operations | umoConfiguration | reports

These properties specify information about reports that are used by IBM Marketing Operations.

reportsAnalysisSectionHome

Description

Indicates the home directory for the Analysis Section reports.

Default value

`/content/folder[@name='Affinium Plan']`

reportsAnalysisTabHome

Description

Indicates the home directory for the Analysis Tab reports.

Default value

`/content/folder[@name='Affinium Plan - Object Specific Reports']`

cacheListOfReports

Description

This parameter enables caching of a list of reports on object instance's analysis page.

Valid values

`True | False`

Default value

`False`

Marketing Operations | umoConfiguration | invoiceRollup

The properties in this category specify options for invoice rollups.

invoiceRollupMode

Description

Specifies how rollups occur. Acceptable values follow.

Valid values

- `immediate`: rollups occur every time that an invoice is marked PAID.
- `schedule`: rollups occur on a scheduled basis.

  If this parameter is set to `schedule`, the system uses the following parameters to determine when rollups occur.
  - `invoiceRollupScheduledStartTime`
  - `invoiceRollupScheduledPollPeriod`

Default value

`immediate`

invoiceRollupScheduledStartTime

Description

If **`invoiceRollupMode`** is `schedule`, this parameter is used as follows.

- If this parameter contains a value (for example, 11:00 pm), that value is the start time for the schedule to start.

- If this parameter is undefined, the rollup schedule starts when the server starts.

If **invoiceRollupMode** is `immediate`, this parameter is not used.

Default value

`11:00 pm`

invoiceRollupScheduledPollPeriod

Description

If **invoiceRollupMode** is `schedule`, this parameter specifies the poll period in seconds for rollup to occur.

If **invoiceRollupMode** is `immediate`, this parameter is not used.

Default value

`3600` (1 hour)

Marketing Operations | umoConfiguration | database

These properties specify information about the database that is used forIBM Marketing Operations.

fileName

Description

Path to file for loading data sources using JNDI lookup.

Default value

`plan_datasources.xml`

sqlServerSchemaName

Description

Specifies the database schema to use. This parameter applies only if you are using SQL Server for your IBM Marketing Operations database.

Default value

`dbo`

db2ServerSchemaName

Important: Changes to the default value supplied for this parameter are not recommended.

Description

Used internally by IBM Marketing Software applications.

Default value

blank

thresholdForUseOfSubSelects

Description

Specifies the number of records beyond which a subquery must be used in the IN clause of SQL (for listing pages) instead of the actual entity IDs in

the IN clause. Setting this parameter improves performance for IBM Marketing Operations installations that have a large application data set. As a best practice, do not change this value unless you encounter performance issues. If this parameter is missing or commented out, the database behaves as if the threshold is set to a large value.

Default value

`3000` (records)

commonDataAccessLayerFetchSize

Description

This parameter specifies resultset fetch size for certain performance sensitive, critical queries.

Default value

`0`

commonDataAccessLayerMaxResultSetSize

Description

This parameter specifies maximum resultset size for certain performance sensitive, critical queries.

Default value

`-1`

useDBSortForAllList

Description

This parameter is used to configure ALL IBM Marketing Operations List Handlers. Use another **useDBSortFor<module>List** parameter to override the paging behavior of a particular list.

Valid values

- `True`: get one page of list data from the database at a time.
- `False`: cache all list data.

Default value

`True`

useDBSortForPlanList

Description

This parameter is used to configure the Plan List Handler.

Valid values

- `True`: get one page of list data from the database at a time.
- `False`: cache all list data.

Default value

`True`

useDBSortForProjectList

Description

This parameter is used to configure the Project List Handler.

Valid values

- `True`: get one page of list data from the database at a time.
- `False`: cache all list data.

Default value

`True`

useDBSortForTaskList

Description

This parameter is used to configure the Task List Handler.

Valid values

- `True`: get one page of list data from the database at a time.
- `False`: cache all list data.

Default value

`True`

useDBSortForProgramList

Description

This parameter is used to configure the Program List Handler.

Valid values

- `True`: get one page of list data from the database at a time.
- `False`: cache all list data.

Default value

`True`

useDBSortForApprovalList

Description

This parameter is used to configure the Approval List Handler.

Valid values

- `True`: get one page of list data from the database at a time.
- `False`: cache all list data.

Default value

`True`

useDBSortForInvoiceList

Description

This parameter is used to configure the Invoice List Handler.

Valid values

- `True`: get one page of list data from the database at a time.
- `False`: cache all list data.

Default value

`True`

useDBSortForAlerts

Description

This parameter is used to configure the Alerts List Handler.

Valid values

- `True`: get one page of list data from the database at a time.
- `False`: cache all list data.

Default value

`True`

Marketing Operations | umoConfiguration | listingPages

These properties specify information about listing items, such as marketing objects or projects, on pages in IBM Marketing Operations.

listItemsPerPage

Description

Specifies how many items (rows) are displayed in one list page. This value must be greater than 0.

Default value

`10`

listPageGroupSize

Description

Specifies the size of visible page numbers in the list navigator in the list page. For example, pages 1-5 is a page group. This value must be greater than 0.

Default value

`5`

maximumItemsToBeDisplayedInCalendar

Description

The maximum number of objects (plans, programs, projects, or tasks) the system displays on calendars. Use this parameter to limit the number of objects that display when users select the calendar view. The number 0 indicates that there is no restriction.

Default value

`0`

listDisplayShowAll

Description

Display "Show All" link on listing pages.

Default value

False

Valid Values

True | False

Marketing Operations | umoConfiguration | objectCodeLocking

These properties specify information about object locks for plans, programs, projects, assets, and marketing objects in IBM Marketing Operations.

enablePersistentObjectLock

Description

This parameter must be set to True if IBM Marketing Operations is deployed in a clustered environment. The object lock information is persistent in the database.

Valid values

True | False

Default value

False

lockProjectCode

Description

Determines whether users can edit the Project Code or PID on the Summary tab of a project.

Valid values

- True: enables locking.
- False: disables locking.

Default value

True

lockProgramCode

Description

Determines whether users can edit the Program Code or PID on the Summary tab of a program.

Valid values

- True: enables locking.
- False: disables locking.

Default value

True

lockPlanCode

Description

Determines whether users can edit the Plan Code or PID on the Plan Summary tab for a plan.

Valid values

- True: enables locking.

- `False`: disables locking.

Default value

`True`

lockMarketingObjectCode

Description

Determines whether users can edit the Marketing Object Code or PID on the Summary tab of a marketing object.

Valid values

- `True`: enables locking.
- `False`: disables locking.

Default value

`True`

lockAssetCode

Description

Determines whether users can edit the Asset Code or PID on the Summary tab of an asset.

Valid values

- `True`: enables locking.
- `False`: disables locking.

Default value

`True`

Marketing Operations | umoConfiguration | thumbnailGeneration

These properties specify information about how and whenIBM Marketing Operations generates thumbnails.

trueTypeFontDir

Description

Specifies the directory where the True Type fonts are located. This parameter is required for thumbnail generation on operating systems other than Windows that use Aspose. For Windows installations, this parameter is optional.

Default value

blank

coreThreadPoolSize

Description

Specifies the number of persistent threads that are kept in the thread pool for thumbnail generator threads.

Default value

5

maxThreadPoolSize

Description

Specifies the maximum number of threads that are allowed in the thread pool for thumbnail generator threads.

Default value

10

threadKeepAliveTime

Description

Parameter to configure the keep-alive time for thumbnail generator threads.

Default value

60

threadQueueSize

Description

Parameter to configure the thread queue size for thumbnail generator threads.

Default value

20

disableThumbnailGeneration

Description

Determines whether thumbnail images are generated for uploaded documents. A value of True enables thumbnail generation.

Default value

False

Valid values

True | False

markupImgQuality

Description

Magnification or zoom factor to apply to the rendered page.

Default value

1

Marketing Operations | umoConfiguration | Scheduler | intraDay

This property specifies how frequently the scheduler runs during the day.

schedulerPollPeriod

Description

Defines how frequently, in seconds, a batch job to calculate project health status runs each day.

Note: Only the daily batch job updates the project health status history, which is used by reports.

Default value

60 (seconds)

Marketing Operations | umoConfiguration | Scheduler | daily

This property specifies what time the scheduler starts each day.

schedulerStartTime

Description

Defines the start time for a batch job that calculates project health status. This job also:

- Updates the project health status history that is used by reports.
- Initiates distribution of email notifications to users who subscribe to them.

Note: The system initiates this batch job only if the calculation is not already running. Define this parameter so that the job starts at a different time than the **intraDay** parameter, and at a time when users are not likely to request this calculation manually.

Default value

11:00 pm

Marketing Operations | umoConfiguration | Notifications

These properties specify information about notifications in IBM Marketing Operations, including information about the event monitor.

notifyPlanBaseURL

Description

The URL for your IBM Marketing Operations deployment, including the host name and port number. Marketing Operations includes this URL in notifications that contain links to other information in Marketing Operations.

Note: Do not use "localhost" as a server name unless your mail client and IBM Marketing Operations server are running on same server.

Default value

http://<*server*>:<*port*>/plan/affiniumplan.jsp

notifyDelegateClassName

Description

The fully qualified Java class name of the delegate implementation to be instantiated by the service. This class must implement the com.unicacorp.afc.service.IServiceImpl interface. Defaults to a local implementation if not specified.

Default value

blank

notifyIsDelegateComplete

Description

Optional Boolean string that indicates whether the delegate implementation is complete. Defaults to `True` if not specified.

Default value

`True`

Valid Values

`True | False`

notifyEventMonitorStartTime

Description

Specifies when the event notification monitor process begins for the first time after IBM Marketing Operations starts. Format the value according to the short version of the `java.text.DateFormat` class for the current locale. For example, in US English locale, a valid string might be `11:45 pm`.

Default value

Blank (Immediately after Marketing Operations is started.)

notifyEventMonitorPollPeriod

Description

Defines the approximate time, in seconds, for the event monitor to sleep between polls. Events accumulate in the event queue between polling periods; shorter polling periods process notifications sooner, but impose more system overhead. If you erase the default and leave the value blank, the poll period defaults to a short period, usually under a minute.

Default value

`5` (seconds)

notifyEventMonitorRemoveSize

Description

Specifies the number of events to remove from the queue in one shot. The event monitor removes events from the event queue in the increments that are specified by this value until none are left.

Note: You can set this value to a number other than 1 to increase event processing performance. However, if the service host goes down before removed events are processed, there is a risk of event loss.

Default value

`10`

alertCountRefreshPeriodInSeconds

Description

Specifies, in seconds, the system-wide alert count refresh period for the alerts count. This count displays near the top of the navigation bar after a user logs in.

Note: Changing the refresh period to poll faster can have performance implications in a multi-user environment.

Default value

180 (3 minutes)

Marketing Operations | umoConfiguration | Notifications | Email

These properties specify information about email notifications in IBM Marketing Operations.

notifyEMailMonitorStartTime

Description

Specifies when the email monitor process runs for the first time after IBM Marketing Operations starts. Format the value according to the short version of the `java.text.DateFormat` class for the current locale. For example, in US English locale, a valid string might be `11:59 pm`.

Default value

Blank (Immediately after IBM Marketing Operations starts.)

notifyEMailMonitorPollPeriod

Description

Defines the approximate time, in seconds, for the email monitor to sleep between polls.

Note: As with events, email messages accumulate in the queue between polling periods; shorter polling times send email messages sooner, but can increase system overhead.

Default value

60 (seconds)

notifyEMailMonitorJavaMailSession

Description

JNDI name of an existing, initialized JavaMail Session to use for email notifications. If not specified and the delegate is marked `Complete`, then the JavaMail host parameter must be supplied so IBM Marketing Operations can create a session.

Default value

blank

notifyEMailMonitorJavaMailProtocol

Description

Specifies the mail server transport protocol to use for email notifications.

Default value

smtp

notifyEMailMonitorRemoveSize

Description

Specifies the number of email messages to remove from queue at one time. The email monitor continues to remove messages from the email queue incrementally until none remain.

Note: You can set this value to a number other than 1 to increase email processing performance. However, if the service host goes down before removed email messages are processed, there is a risk of message loss.

Default value

10 (messages)

notifyEMailMonitorMaximumResends

Description

Specifies the maximum number of times the system attempts to send an email message that failed in the first attempt to send it. When a send fails, the email is put back into the queue until it reaches the maximum number of attempts that are allowed by this parameter.

For example, **notifyEMailMonitorPollPeriod** is set to poll every 60 seconds. Setting the **notifyEMailMonitorMaximumResends** property to 60 attempts causes the email monitor to resend a failed message one time in each poll (every minute), for up to 1 hour. A value of 1440 (24x60) causes the email monitor to try every minute for up to 24 hours.

Default value

1 (attempt)

showUserNameInEmailNotificationTitle

Description

Specifies whether the IBM Marketing Operations notification and alert systems includes the user name in the **From** field of email notifications.

Note: This setting is applicable only to email messages sent by the notification and alert systems in IBM Marketing Operations.

Valid values

- True: Marketing Operations appends the user name to the title of the message and displays both in the **From** field of the email
- False: Marketing Operations displays only the message title in the **From** field

Default value

False

notifyEMailMonitorJavaMailDebug

Description

Specifies whether JavaMail debug mode is set.

Valid values

- `True`: enables JavaMail debug.
- `False`: disables debug tracing.

Default value

`False`

Marketing Operations | umoConfiguration | Notifications | project

These properties specify information about project alarms in IBM Marketing Operations.

notifyProjectAlarmMonitorStartTime

Description

Specifies when the project alarm monitors run for the first time after IBM Marketing Operations starts. Format the value according to the short version of the `java.text.DateFormat` class for the current locale. For example, in US English locale, a valid string might be `11:59 pm`. If you erase the default and leave the value blank, this monitor starts immediately after you create it.

Default value

`10:00 pm`

notifyProjectAlarmMonitorPollPeriod

Description

Defines the approximate time, in seconds, for the project and program alarm monitors to sleep between polls.

Default value

Blank (60 seconds)

notifyProjectAlarmMonitorScheduledStartCondition

Description

Defines the number of days before the start date of a project for IBM Marketing Operations to send notifications to users.

Note: If this value is `-1`, then Marketing Operations does not send these notifications.

Default value

1 (day)

notifyProjectAlarmMonitorScheduledEndCondition

Description

Defines the number of days before the end date of a project for IBM Marketing Operations to send end notifications to users.

Note: If this value is `-1`, then Marketing Operations does not send these notifications.

Default value

3 (days)

notifyProjectAlarmMonitorTaskScheduledStartCondition

Description

Defines the number of days before the start date of a task for IBM Marketing Operations to send start notifications to users.

Note: If this value is -1, then Marketing Operations does not send these notifications.

Default value

1 (day)

notifyProjectAlarmMonitorTaskScheduledEndCondition

Description

Defines the number of days before the end date of a task for IBM Marketing Operations to send end notifications to users.

Note: If this value is -1, then Marketing Operations does not send these notifications.

Default value

3 (days)

notifyProjectAlarmMonitorTaskLateCondition

Description

Defines the number of days after the start date of a task for IBM Marketing Operations to send users notification that a task did not start.

Note: If this value is -1, then Marketing Operations does not send these notifications.

Default value

3 (days)

notifyProjectAlarmMonitorTaskOverdueCondition

Description

Defines the number of days after the end date of a task for IBM Marketing Operations to send users notification that a task did not finish.

Note: If this value is -1, then Marketing Operations does not send these notifications.

Default value

3 (days)

notifyProjectAlarmMonitorTaskScheduledMilestoneCondition

Description

Defines the number of days before the start date of a milestone task for IBM Marketing Operations to send notifications.

Note: If this value is -1, then Marketing Operations does not send these notifications.

Default value

1 (day)

Marketing Operations | umoConfiguration | Notifications | projectRequest

These properties specify information about project request alarms in IBM Marketing Operations.

notifyRequestAlarmMonitorLateCondition

Description

Defines the number of days for IBM Marketing Operations to send a notification that the request is late.

Note: If this value is -1, then Marketing Operations does not send these notifications.

Default value

3 (days)

notifyRequestAlarmMonitorScheduledEndCondition

Description

Defines the number of days before the end date of a request for IBM Marketing Operations to send end notifications to users.

Note: If this value is -1, then Marketing Operations does not send these notifications.

Default value

1 (day)

Marketing Operations | umoConfiguration | Notifications | program

The properties in this category specify options for program notification schedules.

notifyProgramAlarmMonitorScheduledStartCondition

Description

Defines the number of days before the start date of a program that IBM Marketing Operations sends start notifications to users.

Note: If this value is -1, then Marketing Operations does not send these notifications.

Default value

1 (day)

notifyProgramAlarmMonitorScheduledEndCondition

Description

Defines the number of days before the end date of a program thatIBM Marketing Operations sends end notifications to users.

Note: If this value is -1, then Marketing Operations does not send these notifications.

Default value

3 (days)

Marketing Operations | umoConfiguration | Notifications | marketingObject

These properties specify information about marketing object alarms in IBM Marketing Operations.

notifyComponentAlarmMonitorScheduledStartCondition

Description

Specifies the number of days before the start date of a marketing object for IBM Marketing Operations to send start notifications to users.

Note: If this value is -1, then Marketing Operations does not send these notifications.

Default value

1 (day)

notifyComponentAlarmMonitorScheduledEndCondition

Description

Specifies the number of days before the end date of a marketing object for IBM Marketing Operations to send end notifications to users.

Note: If this value is -1, then Marketing Operations does not send these notifications.

Default value

3 (days)

Marketing Operations | umoConfiguration | Notifications | approval

These properties specify information about approval alarms in IBM Marketing Operations.

notifyApprovalAlarmMonitorStartTime

Description

Specifies when the approval alarm monitor begins processing for the first time after IBM Marketing Operations starts. Format the value according to the short version of the `java.text.DateFormat` class for the current locale. For example, in US English locale, a valid string might be `11:59 pm`. If you delete the default and leave this value blank, the monitor starts immediately after it is created.

Note: For best results, configure the alarm monitors to start during off-peak hours and stagger their start times to spread out the data processing load.

Default value

9:00 pm

notifyApprovalAlarmMonitorPollPeriod

Description

Specifies the approximate time, in seconds, for the approval alarm monitor to sleep between polls.

Default value

Blank (60 seconds)

notifyApprovalAlarmMonitorLateCondition

Description

Specifies the number of days after the start date of an approval for the system to begin notifying users that the approval is late.

Note: If this value is -1, then Marketing Operations does not send these notifications.

Default value

3 (days)

notifyApprovalAlarmMonitorScheduledEndCondition

Description

Specifies the number of days before the end date of an approval for the system to begin sending end notifications to users.

Note: If this value is -1, then Marketing Operations does not send these notifications.

Default value

1 (day)

Marketing Operations | umoConfiguration | Notifications | asset

These properties specify information about asset alarms in IBM Marketing Operations.

notifyAssetAlarmMonitorStartTime

Description

Specifies when the asset alarm monitor process runs for the first time after IBM Marketing Operations starts. Format the value according to the short version of the java.text.DateFormat class for the current locale. For example, in US English locale, a valid string might be 11:59 pm. If you delete the default and leave this value blank, the monitor starts immediately after it is created.

Note: For best results, configure the alarm monitors to start during off-peak hours and stagger their start times to spread out the data processing load.

Default value

11:00 pm

notifyAssetAlarmMonitorPollPeriod

Description

Specifies the time, in seconds, for the asset alarm monitor to sleep between polls.

Default value

Blank (60 seconds)

notifyAssetAlarmMonitorExpirationCondition

Description

Specifies the number of days before an asset is going to expire for IBM Marketing Operations to notify users that the asset is about to expire.

Note: If this value is -1, Marketing Operations does not check for expiration.

Default value

5 (days)

Marketing Operations | umoConfiguration | Notifications | invoice

These properties specify information about invoice alarms in IBM Marketing Operations.

notifyInvoiceAlarmMonitorStartTime

Description

Specifies when the invoice alarm monitor process runs for the first time after IBM Marketing Operations starts. Format the value according to the short version of the java.text.DateFormat class for the current locale. For example, in US English locale, a valid string might be 11:59 pm. If you delete the default and leave the value blank, the monitor starts immediately after you create it.

Note: For best results, configure the alarm monitors to start during off-peak hours and to stagger their start times to spread out the data processing load.

Default value

9:00 pm

notifyInvoiceAlarmMonitorDueCondition

Description

Specifies the number of days before the due date for IBM Marketing Operations to notify users that an invoice is due.

Note: If this value is -1, then Marketing Operations does not send these notifications.

Default value

5 (days)

Campaign configuration properties

The configuration properties for IBM Campaign are located at **Settings** > **Configuration**.

Campaign

To specify the locales and component applications that your installation supports, choose **Settings** > **Configuration**, then click the Campaign category.

currencyLocale

Description

The currencyLocale property is a global setting that controls how currency is displayed in the Campaign web application, regardless of the display locale.

Important: No currency conversion is performed by Campaign when the display locale changes (for example, if the multi-locale feature is implemented and the display locale changes based on user-specific locales). You must be aware that when a locale is switched, for example, from English US, in which a currency amount is, for example, US$10.00, to a French locale, the currency amount is unchanged (10,00) even if the currency symbol changes with the locale.

Default value

en_US

supportedLocales

Description

The supportedLocales property specifies the locales or language-locale pairs that Campaign supports. The value of this property is set by the installer when you install Campaign. For example: de,en,fr,ja,es,ko,pt,it,zh,ru.

Default value

All languages/locales into which Campaign has been localized.

defaultLocale

Description

The defaultLocale property specifies which of the locales specified in the supportedLocales property is considered the default display locale for Campaign. The value of this property is set by the installer when you install Campaign.

Default value

en

acoInstalled

Path

Description

The acoInstalled property specifies whether Contact Optimization is installed.

When Contact Optimization is installed and configured, set the value to yes, which causes the Contact Optimization process to be displayed in flowcharts. If the value is true and Contact Optimization is not installed or configured, the process is displayed but disabled (grayed out).

Default value

false

Valid Values

false and true

collaborateInstalled

Description

The collaborateInstalled property specifies whether Distributed Marketing is installed. When Distributed Marketing is installed and configured, set the value to true, which causes the Distributed Marketing features to be available in the Campaign user interface.

Default value

false

Valid Values

true | false

Campaign | collaborate

The properties in this category pertain to Distributed Marketing configuration.

CollaborateIntegrationServicesURL

Description

The CollaborateIntegrationServicesURL property specifies the server and port number of Distributed Marketing. This URL is used by Campaign when a user publishes a flowchart to Distributed Marketing.

Default value

http://localhost:7001/collaborate/services/
CollaborateIntegrationServices1.0

Campaign | navigation

Some of the properties in this category are used internally and should not be changed.

welcomePageURI

Configuration category

Campaign|navigation

Description

The welcomePageURI property is used internally by IBM applications. It specifies the Uniform Resource Identifier of the Campaign index page. You should not change this value.

Default value

No default value defined.

seedName

Configuration category

Campaign|navigation

Description

The seedName property is used internally by IBM applications. You should not change this value.

Default value

No default value defined.

type

Configuration category

Campaign|navigation

Description

The type property is used internally by IBM applications. You should not change this value.

Default value

No default value defined.

httpPort

Configuration category

Campaign|navigation

Description

This property specifies the port used by the Campaign web application server. If your installation of Campaign uses a port that is different from the default, you must edit the value of this property.

Default value

7001

httpsPort

Configuration category

Campaign|navigation

Description

If SSL is configured, this property specifies the port used by the Campaign web application server for secure connections. If your installation of Campaign uses a secure port that is different from the default, you must edit the value of this property.

Default value

7001

serverURL

Configuration category

Campaign|navigation

Description

The serverURL property specifies the URL used by Campaign. If your installation of Campaign has a URL that is different from the default, you should edit the value as follows:

http://*machine_name_or_IP_address:port_number/context-root*

If users access Campaign with the Chrome browser, use the fully qualified domain name (FQDN). If the FQDN is not used, the Chrome browser cannot access the product URLs.

Default value

http://localhost:7001/Campaign

logoutURL

Configuration category

Campaign|navigation

Description

The logoutURL property is used internally to call the logout handler of the registered application if the user clicks the logout link. Do not change this value.

serverURLInternal

Configuration category

Campaign|navigation

Description

The serverURLInternal property specifies the URL for the Campaign web application when SiteMinder is used; this property is also used for internal communication with other IBM Marketing Software applications, such as eMessage and Interact. If the property is empty, the value in the serverURL property is used. Modify this property if you need internal application communication to be http and external communication to be https. If you use SiteMinder, you must set this value to the URL for the Campaign web application server, formatted as follows:

http://*machine_name_or_IP_address:port_number/context-root*

Default value

No default value defined.

campaignDetailPageURI

Configuration category

Campaign|navigation

Description

The campaignDetailPageURI property is used internally by IBM applications. It specifies the Uniform Resource Identifier of the Campaign detail page. You should not change this value.

Default value

campaignDetails.do?id=

flowchartDetailPageURI

Configuration category

Campaign|navigation

Description

The flowchartDetailPageURI property is used to construct a URL to navigate to the details of a flowchart in a specific campaign. You should not change this value.

Default value

flowchartDetails.do?campaignID=&id=

schedulerEditPageURI

Configuration category

Campaign|navigation

Description

This property is used to construct a URL to navigate to the Scheduler page. Do not change this value.

Default value

jsp/flowchart/scheduleOverride.jsp?taskID=

offerDetailPageURI

Configuration category

Campaign|navigation

Description

The offerDetailPageURI property is used to construct a URL to navigate to the details of a specific offer. You should not change this value.

Default value

offerDetails.do?id=

offerlistDetailPageURI

Configuration category

Campaign|navigation

Description

The offerlistDetailPageURI property is used to construct a URL to navigate to the details of a specific offer list. You should not change this value.

Default value

displayOfferList.do?offerListId=

mailingDetailPageURI

Configuration category

Campaign|navigation

Description

This property is used to construct a URL to navigate to the mailing details page for eMessage. Do not change this value.

Default value

view/MailingDetails.do?mailingId=

optimizeDetailPageURI

Configuration category

Campaign|navigation

Description

This property is used to construct a URL to navigate to IBM Contact Optimization details page. Do not change this value.

Default value

optimize/sessionLinkClicked.do?optimizeSessionID=

optimizeSchedulerEditPageURI

Configuration category

Campaign|navigation

Description

This property is used to construct a URL to navigate to the IBM Contact Optimization Scheduler edit page. Do not change this value.

Default value

optimize/editOptimizeSchedule.do?taskID=

displayName

Configuration category

Campaign|navigation

Description

The displayName property specifies the link text used for the Campaign link in the drop-down menu that exists in the GUI of each IBM product.

Default value

Campaign

Campaign | caching

Certain objects, such as offers, are cached in the web application server to improve response times in the IBM Campaign user interface. The Campaign|caching configuration properties specify the length of time that cached data is retained. Smaller values result in more frequent cache updates, which can adversely affect performance by consuming processing resources on both the web server and the database.

offerTemplateDataTTLSeconds

Configuration category

Campaign|caching

Description

The offerTemplateDataTTLSeconds property specifies the length of time, in seconds, that the system retains the Offer Template cache data (Time to Live). An empty value means the cache data is never purged.

Default value

600 (10 minutes)

campaignDataTTLSeconds

Configuration category

Campaign|caching

Description

The campaignDataTTLSeconds property specifies the length of time, in seconds, that the system retains the Campaign cache data (Time to Live). An empty value means the cache data is never purged.

Default value

600 (10 minutes)

sessionDataTTLSeconds

Configuration category

Campaign|caching

Description

The sessionDataTTLSeconds property specifies the length of time, in seconds, that the system retains the Session cache data (Time to Live). An empty value means the cache data is never purged.

Default value

600 (10 minutes)

folderTreeDataTTLSeconds

Configuration category

Campaign|caching

Description

The folderTreeDataTTLSeconds property specifies the length of time, in seconds, that the system retains the Folder Tree cache data (Time to Live). An empty value means the cache data is never purged.

Default value

600 (10 minutes)

attributeDataTTLSeconds

Configuration category

Campaign|caching

Description

The attributeDataTTLSeconds property specifies the length of time, in seconds, that the system retains the Offer Attribute cache data (Time to Live). An empty value means the cache data is never purged.

Default value

600 (10 minutes)

initiativeDataTTLSeconds

Configuration category

Campaign|caching

Description

The initiativeDataTTLSeconds property specifies the length of time, in seconds, that the system retains the Initiative cache data (Time to Live). An empty value means the cache data is never purged.

Default value

600 (10 minutes)

offerDataTTLSeconds

Configuration category

Campaign|caching

Description

The offerDataTTLSeconds property specifies the length of time, in seconds, that the system retains the Offer cache data (Time to Live). An empty value means the cache data is never purged.

Default value

600 (10 minutes)

segmentDataTTLSeconds

Configuration category

Campaign|caching

Description

The segmentDataTTLSeconds property specifies the length of time, in seconds, that the system retains the Segment cache data (Time to Live). An empty value means the cache data is never purged.

Default value

600 (10 minutes)

Campaign | partitions

This category contains properties to configure IBM Campaign partitions, including the default partition, which is named partition1.

One category should be created for each Campaign partition. This section describes the properties in the partition[n] category, which apply to all partitions that you configure in Campaign.

Campaign | partitions | partition[n] | eMessage

Define properties in this category to define characteristics of recipient lists and specify the location of resources that upload the lists to IBM Marketing Software Hosted Services.

eMessagePluginJarFile

Description

Complete path to the location of the file that operates as the Recipient List Uploader (RLU). This plug-in to Campaign uploads OLT data and associated metadata to the remote services hosted by IBM. The location that you specify must be the full local directory path in the file system for the computer that hosts the Campaign web application server.

The IBM installer populates this setting automatically for the default partition when you run the installer. For other partitions, you must configure this property manually. Because there is only one RLU for each eMessage installation, all partitions must specify the same location for the RLU.

Do not change this setting unless IBM instructs you to do so.

Default value

No default value defined.

Valid Values

Full local directory path to where you installed the Campaign web server.

defaultSeedInterval

Description

The number of messages between seed messages if `defaultSeedType` is `Distribute list`.

Default value

`1000`

defaultSeedType

Description

The default method that eMessage uses to insert seed addresses into a recipient list.

Default value

`Distribute IDS`

Valid Values

- `Distribute IDS` - Distribute IDs evenly, based on the size of the recipient list and the number of seed addresses available, inserts seed addresses at equal intervals throughout the entire recipient list.
- `Distribute list` - Insert seed address for every `defaultSeedInterval` IDs in main list. Inserts the entire list of available seed addresses at specified intervals throughout the recipient list. You must specify the interval between insertion points.

oltTableNamePrefix

Description

Used in the generated schema for the output list table. You must define this parameter.

Default value

OLT

Valid Values

The prefix can contain no more than 8 alphanumeric or underscore characters, and must start with a letter.

oltDimTableSupport

Description

This configuration parameter controls the ability to add dimension tables to output list tables (OLT) created in the eMessage schema. Dimension tables are required to use advanced scripting for email to create data tables in email messages.

The default setting is `False`. You must set this property to `True` so that marketers can create dimension tables when they use the eMessage process to define a recipient list. For more information about creating data tables and working with advanced scripts for email, see the *IBM eMessage User's Guide.*

Default value

False

Valid Values

True | False

Campaign | partitions | partition[n] | eMessage | contactAndResponseHistTracking

Use the properties in this category to configure eMessage offer integration with IBM Campaign for the current partition.

etlEnabled

Description

Campaign uses its own ETL process to extract, transform, and load offer response data from the eMessage tracking tables into the Campaign contact and response history tables.

The ETL process coordinates information across the necessary tables, including `UA_UsrResponseType` (Campaign response types) and `UA_RespTypeMapping` (mapping of response types between Campaign and eMessage).

Setting the value to `Yes` ensures that information about eMessage offer contact and response history is coordinated between Campaign and eMessage. For example, email response data will be included in Campaign reports.

Note: You must also set `Campaign | partitions | partition[n] | server | internal | eMessageInstalled` to `Yes` for this partition or the ETL process will not run.

Default value

`No`

Valid values

`Yes | No`

runOnceADay

Description

Indicate whether the ETL process should run only once a day.

If the value is `Yes`: You must specify a **startTime**; the ETL job then runs until all of the records are processed; and the **sleepIntervalInMinutes** is ignored.

If the value is `No`: The ETL job starts as soon as the Campaign web server starts. The ETL job stops after all of the records are processed, then waits for the time specified by **sleepIntervalInMinutes**.

Default value

`No`

Valid values

`Yes | No`

batchSize

Description

The ETL process uses this parameter to fetch records that have been downloaded by the RCT into the local eMessage system tables. Because large values can impact performance, the list of available values is restricted to the valid values shown below. If you anticipate large volumes of records, adjust the **batchSize** together with the **sleepIntervalInMinutes** to process records at regular intervals.

Default value

`100`

Valid values

100 | 200 | 500 | 1000

sleepIntervalInMinutes

Description

Specify the interval in minutes between ETL jobs. This option determines the wait time after a job finishes. The ETL process waits for this duration before starting the next job. Multiple jobs can run synchronously and there may be multiple ETL jobs per partition.

If **runOnceADay** is `Yes`, you cannot set a sleep interval.

Default value

`60`

Valid values

Positive integers

startTime

Description

Specify a time to start the ETL job. You must use the English locale format to specify the start time.

Default value

12:00:00 AM

Valid values

Any valid time in the format hh:mm:ss AM/PM

notificationScript

Description

An optional executable or script file that is run after each ETL job is done. For example, you might want to be notified of the success or failure of each ETL job, for monitoring purposes. The notification script runs every time the ETL job for a given partition finishes running.

The parameters passed in to this script are fixed and cannot be changed. The following parameters can be used by the script:

- etlStart: The start time of ETL in number of milliseconds.
- etlEnd: The end time of ETL in number of milliseconds.
- totalCHRecords: Total number of contact records processed.
- totalRHRecords: Total number of response histroy records processed.
- executionStatus: Execution status of the ETL with value 1 (failed) or 0 (succeeded).

Default value

No default value defined.

Valid values

Any valid path that the Campaign server can access with Read and Execute permissions. For example: D:\myscripts\scriptname.exe

Campaign | partitions | partition[n] | Engage

These properties control authentication and data exchange between IBM Campaign and IBM Engage if the products are integrated.

To access these properties, choose **Settings** > **Configuration**. If your Campaign installation has multiple partitions, set these properties for each partition that uses the integration.

Service URL

Configuration category

Campaign | partitions | partition[n] | Engage

Description

The Service URL indicates the URL where Campaign can access the IBM Engage application. The Engage Org Admin must provide this value.

Default value

<none>

Example

https://engageapi.abc01.com/

OAuth URL Suffix

Configuration category

Campaign | partitions | partition[n] | Engage

Description

The `OAuth URL Suffix` specifies the authentication token for the Engage APIs.

Default value

oauth/token

API URL Suffix

Configuration category

Campaign | partitions | partition[n] | Engage

Description

The `API URL Suffix` is set to `XMLAPI` to ensure that Campaign uses the Engage XML APIs. Best practice is to leave this set to the default value.

Default value

XMLAPI

Platform User with Data Sources for Engage Credentials

Configuration category

Campaign | partitions | partition[n] | Engage

Description

The `Platform User with Data Sources for Engage Credentials` indicates the name of the IBM Marketing Platform user account that is allowed to connect to the IBM Engage server. This user account contains the data sources that provide Engage credentials. Typically, `asm_admin` is used.

Default value

No default value defined.

Valid values

The IBM Marketing Platform user account that contains the data sources for Engage integration credentials.

Data Source for Client ID

Configuration category

Campaign | partitions | partition[n] | Engage

Description

The `Data Source for Client ID` value must exactly match the name of the Engage Client ID data source that was created for the user account that connects to the IBM Engage server (**Platform User with Data Sources for Engage Credentials**). In other words, the value must match what is set up as the datasource for the IBM Marketing Platform user. Best practice is to leave this set to the default value.

Default value

ENGAGE_CLIENT_ID_DS

Data Source for Client Secret

Configuration category

Campaign | partitions | partition[n] | Engage

Description

The Data Source for Client Secret value must exactly match the name of the Engage Client Secret data source that was created for the user account that connects to the IBM Engage server (**Platform User with Data Sources for Engage Credentials**). Best practice is to leave this set to the default value.

Default value

ENGAGE_CLIENT_SECRET_DS

Data Source for Client Refresh Token

Configuration category

Campaign | partitions | partition[n] | Engage

Description

The Data Source for Client Refresh Token value must exactly match the name of the Engage Client Refresh Token data source that was created for the user account that connects to the IBM Engage server (**Platform User with Data Sources for Engage Credentials**). Best practice is to leave this set to the default value.

Default value

ENGAGE_CLIENT_REF_TOK_DS

Data Source for File Transfer Credentials

Configuration category

Campaign | partitions | partition[n] | Engage

Description

The Data Source for File Transfer Credentials indicates the name of the data source that provides the credentials for FTP communication between Campaign and Engage. This value must exactly match the name of the Engage FTP data source that was created for the user account that connects to the IBM Engage server (**Platform User with Data Sources for Engage Credentials**). Best practice is to leave this set to the default value.

Default value

ENGAGE_FTP_DS

Host Name for File Transfer

Configuration category

Campaign | partitions | partition[n] | Engage

Description

The Host Name for File Transfer indicates the host name of the Engage FTP server where Campaign uploads the contact list in TSV format. This file gets deleted automatically after it is uploaded into a contact list.

Default value

<none>

Valid values

Any valid address in the list of IBM Marketing Cloud FTP addresses: http://www.ibm.com/support/knowledgecenter/SSTSRG/Setting_up_an_FTP_or_SFTP_account.html?lang=en. For example: transfer2.silverpop.com

Port Number for File Transfer

Configuration category

Campaign | partitions | partition[n] | Engage

Description

The `Port Number for File Transfer` indicates the port number for the FTP server that is specified in **Host Name for File Transfer**.

Default value

22

Valid values

Any valid FTP port number

Use proxy for ServiceURL

Description

Determine if you use proxy for ServiceURL. If you select `Yes`, your connection uses the proxy server. Proxy server details can be configured under Campaign | proxy. If you select `No`, a proxy server is not used to connect to Engage.

Default value

No

Valid values

Yes, No

Use proxy for FTP

Description

Determine if you use proxy for FTP. If you select `Yes`, your connection to the Engage FTP server uses the proxy server. Proxy server details can be configured under Campaign | proxy. If you select `No`, a proxy server is not used to connect to the Engage FTP server.

Default value

No

Valid values

Yes, No

allowableFailurePercentage

Description

When a campaign contact list is processed in Engage, an error is returned to Campaign if a contact fails for any reason. Use the `allowableFailurePercentage` property to specify the percentage of failed records that are allowed before the campaign process fails. If the percentage of failed records is greater than the configured `allowableFailurePercentage`, the process fails.

If the percentage of failed records is less than or equal to the configured `allowableFailurePercentage`, the process does not fail. Duplicate records are considered as valid records and so they do not affect the `allowableFailurePercentage` for a process box. All valid records are imported into Engage.

The percentage of failed records is calculated as `([TOTAL_ROWS - TOTAL_VALID]/ TOTAL_ROWS) *100`.

Any warnings and errors for the process are logged in the ImportErrors log file. You can delete this file at regular intervals. The ImportErrors log file is located in the <Campaign_home>/logs folder.

Default value
0

Valid values
Integers between 0-100.

Campaign | partitions | partition[n] | Engage | contactAndResponseHistTracking

These properties specify the ETL of events that are downloaded from UBX in the Campaign history tables.

To access these properties, choose **Settings** > **Configuration**. If your Campaign installation has multiple partitions, set these properties for each partition that uses the integration.

etlEnabled

Description
Determine whether you want to enable the ETL transfer of data from the events table in the Campaign history table.

Default value
No

Valid values
Yes, No

runOnceADay

Description
Determine whether the ETL runs once a day. It can run repeatedly if you specify the sleepIntervalInMinutes property. If *runOnceADay* is set to yes, ETL runs once a day at the specified time.

Valid values
Yes, No

batchSize

Description
The number of records that are processed in one ETL cycle.

Default value
100

Valid values
100, 200, 500, 1000, 10000, 100000

sleepIntervalInMinutes

Description
Specify the number of minutes the ETL waits it runs again. This value is used when runOnceADay is set to No.

Default value
60

Valid values

Positive integers.

startTime

Description

When runOnceADay is set to Yes, this property determines the ETL run start.

Default value

`12:00:00 AM`

Valid values

Any valid time in the format `hh:mm:ss AM/PM`.

notificationScript

Description

Enter any script that can run after the ETL execution is complete.

Default value

No default value defined.

Valid values

Any valid path that the Campaign server can access with Read and Execute permissions. Example: `D:\myscripts\scriptname.exe`

Campaign | partitions | partition[n] | UBX

These properties control authentication and data exchange between IBM Campaign, IBM Engage, and IBM UBX if the products are integrated.

To access these properties, choose **Settings** > **Configuration**. If your Campaign installation has multiple partitions, set these properties for each partition that uses the integration.

API URL

Description

Specify the UBX Server API URL.

Data Source for UBX Endpoint Authorization key

Description

Specify the datasource name that contains the authorization key for the Campaign registered endpoint. For example, `UBX_DS`.

Platform User with Data Sources for UBX Credentials

Description

Specify the Marketing Platform user name that contains the datasource with the name specified under the configuration property **Data Source for UBX Endpoint Authorization key**.

Use proxy for API URL

Description

Determine if you want to use a proxy server for the UBX connection. If you select `Yes`, the proxy server details are configured under Campaign | proxy.

Campaign | partitions | partition[n] | UBX | Event Download Schedule

These properties specify the schedule for when events are downloaded from UBX into Campaign.

To access these properties, choose **Settings** > **Configuration**. If your Campaign installation has multiple partitions, set these properties for each partition that uses the integration.

Event Download Enabled

Description

Determine if you want to enable events from UBX to download to the events table in the Campaign system schema.

Default value

No

Valid values

Yes, No

runOnceADay

Description

Determine if the download should occur on a daily basis. It can run repeatedly if you specify the sleepIntervalInMinutes property.

sleepIntervalInMinutes

Description

Specify the number of minutes the download waits before executing again. This value is used when runOnceADay is set to No.

startTime

Description

When runOnceADay is set to Yes, this property determines when the event download starts.

Campaign | partitions | partition[n] | Coremetrics

The properties in this category specify integration settings for Digital Analytics and Campaign for the selected partition.

If your Campaign installation has multiple partitions, set these properties for each partition that you want to affect. For these properties to take effect, `UC_CM_integration` must be set to `Yes` for the partition (under `partitions | partition[n] | server | internal`).

ServiceURL

Description

The `ServiceURL` specifies the location of the Digital Analytics integration service that provides the integration point between Digital Analytics and Campaign. Note that the default port for https is 443.

Default value

`https://export.coremetrics.com/eb/segmentapi/1.0/api.do`

Valid values

The only supported value for this release is the default value shown above.

CoremetricsKey

Description

Campaign uses the `CoreMetricsKey` to map IDs exported from Digital Analytics to the corresponding Audience ID in Campaign. The value defined for this property must exactly match the value used in the translation table.

Default value

`registrationid`

Valid values

The only supported value for this release is `registrationid`.

ClientID

Description

Set this value to the unique Digital Analytics Client ID assigned to your company.

Default value

No default value defined.

TranslationTableName

Description

Specify the name of the translation table being used to translate Digital Analytics keys to Campaign Audience IDs. For example, `Cam_CM_Trans_Table`. If you do not specify a table name, an error will occur if users run a flowchart that uses Digital Analytics segments as input, because without the table name, Campaign does not know how to map IDs from one product to the other.

Note: When you map or re-map a translation table, the **IBM Table Name** assigned in the Table Definition dialog must exactly match (including case) the `TranslationTableName` defined here.

Default value

No default value defined.

ASMUserForCredentials

Description

The `ASMUserForCredentials` property specifies which IBM Marketing Software account is allowed to access the Digital Analytics integration service. See below for additional information.

If no value is specified, Campaign checks the currently logged-in user's account to see if the `ASMDatasourceForCredentials` value is specified as a data source. If it is, then access is allowed. If not, access is denied.

Default value

`asm_admin`

ASMDataSourceForCredentials

Description

The ASMDataSourceForCredentials property identifies the data source assigned to the Marketing Platform account specified in the **ASMUserForCredentials** setting. The default is UC_CM_ACCESS. This "data source for credentials" is the mechanism that Marketing Platform uses to store the credentials that provide access to the integration service.

Although a default value of UC_CM_ACCESS is supplied, a data source of that name is not provided, nor do you have to use that name.

Important: You must choose **Settings > Users**, select the user specified in ASMUserForCredentials, click the **Edit Data Sources** link, and add a new data source whose name exactly matches the value defined here (for example,UC_CM_ACCESS). For Data Source Login and Data Source Password, use the credentials associated with your Digital Analytics Client ID. For information about data sources, user accounts, and security, see the *IBM Marketing Platform Administrator's Guide*

Default value

UC_CM_ACCESS

Related tasks:

"Next steps" on page 116

Campaign | partitions | partition[n] | reports

The **Campaign | partitions | partition[n] | reports** property defines the different types of folders for reports.

offerAnalysisTabCachedFolder

Description

The offerAnalysisTabCachedFolder property specifies the location of the folder that contains the specification for bursted (expanded) offer reports listed on the Analysis tab when you reach it by clicking the Analysis link on the navigation pane. The path is specified by using the XPath notation.

Default value

```
/content/folder[@name='Affinium Campaign - Object Specific
Reports']/folder[@name='offer']/folder[@name='cached']
```

segmentAnalysisTabOnDemandFolder

Description

The segmentAnalysisTabOnDemandFolder property specifies the location of the folder that contains the segment reports listed on the Analysis tab of a segment. The path is specified by using the XPath notation.

Default value

```
/content/folder[@name='Affinium Campaign - Object Specific
Reports']/folder[@name='segment']/folder[@name='cached']
```

offerAnalysisTabOnDemandFolder

Description

The offerAnalysisTabOnDemandFolder property specifies the location of the folder that contains the offer reports listed on the Analysis tab of an offer. The path is specified by using the XPath notation.

Default value

```
/content/folder[@name='Affinium Campaign - Object Specific
Reports']/folder[@name='offer']
```

segmentAnalysisTabCachedFolder

Description

The segmentAnalysisTabCachedFolder property specifies the location of the folder that contains the specification for bursted (expanded) segment reports listed on the Analysis tab when you reach it by clicking the Analysis link on the navigation pane. The path is specified by using the XPath notation.

Default value

```
/content/folder[@name='Affinium Campaign - Object Specific
Reports']/folder[@name='segment']
```

analysisSectionFolder

Description

The analysisSectionFolder property specifies the location of the root folder where report specifications are stored. The path is specified by using the XPath notation.

Default value

```
/content/folder[@name='Affinium Campaign']
```

campaignAnalysisTabOnDemandFolder

Description

The campaignAnalysisTabOnDemandFolder property specifies the location of the folder that contains the campaign reports listed on the Analysis tab of a campaign. The path is specified by using the XPath notation.

Default value

```
/content/folder[@name='Affinium Campaign - Object Specific
Reports']/folder[@name='campaign']
```

campaignAnalysisTabCachedFolder

Description

The campaignAnalysisTabCachedFolder property specifies the location of the folder that contains the specification for bursted (expanded) campaign reports listed on the Analysis tab when you reach it by clicking the Analysis link on the navigation pane. The path is specified by using the XPath notation.

Default value

/content/folder[@name='Affinium Campaign - Object Specific Reports']/folder[@name='campaign']/folder[@name='cached']

campaignAnalysisTabEmessageOnDemandFolder

Description

The campaignAnalysisTabEmessageOnDemandFolder property specifies the location of the folder that contains the eMessage reports listed on the Analysis tab of a campaign. The path is specified by using the XPath notation.

Default value

/content/folder[@name='Affinium Campaign']/folder[@name='eMessage Reports']

campaignAnalysisTabInteractOnDemandFolder

Description

Report server folder string for Interact reports.

Default value

/content/folder[@name='Affinium Campaign']/folder[@name='Interact Reports']

Availability

This property is applicable only if you install Interact.

interactiveChannelAnalysisTabOnDemandFolder

Description

Report server folder string for Interactive Channel analysis tab reports.

Default value

/content/folder[@name='Affinium Campaign - Object Specific Reports']/folder[@name='interactive channel']

Availability

This property is applicable only if you install Interact.

Campaign | partitions | partition[n] | validation

The Validation Plugin Development Kit (PDK), delivered with Campaign, allows third parties to develop custom validation logic for use in Campaign. Properties in the partition[n] > validation category specify the classpath and class name of the custom validation program, and an optional configuration string.

validationClass

Description

The validationClass property specifies the name of the class used for validation in Campaign. The path to the class is specified in the validationClasspath property. The class must be fully qualified with its package name.

For example:

com.unica.campaign.core.validation.samples.SimpleCampaignValidator

indicates the SimpleCampaignValidator class from the sample code.

This property is undefined by default, which causes Campaign to perform no custom validation.

Default value

No default value defined.

validationConfigString

Description

The validationConfigString property specifies a configuration string that is passed into the validation plugin when Campaign loads it. The use of the configuration string may vary, depending on the plugin used.

This property is undefined by default.

Default value

No default value defined.

validationClasspath

Description

The validationClasspath property specifies the path to the class used for custom validation in Campaign.

- Use either a full path or a relative path. If the path is relative, the behavior depends on the application server that is running Campaign. WebLogic uses the path to the domain work directory, which by default is c:\bea\user_projects\domains\mydomain.
- If the path ends in a slash (forward slash / for UNIX or backslash \ for Windows), Campaign assumes that it points to the location of the Java plug-in class that should be used.
- If the path does not end in a slash, Campaign assumes that it is the name of a .jar file that contains the Java class. For example, the value /<*CAMPAIGN_HOME*>/devkits/validation/lib/validator.jar is the path on a UNIX platform that points to the JAR file that is provided with the plug-in developer's kit.

This property is undefined by default, which causes the property to be ignored.

Default value

No default value defined.

Campaign | partitions | partition[n] | audienceLevels | audienceLevel

Do not edit properties in this category. These properties are created and populated when a user creates audience levels on the Administration page in Campaign.

numFields

Description

This property indicates the number of fields in the audience level. Do not edit this property.

Default value

No default value defined.

audienceName

Description

This property indicates the audience name. Do not edit this property.

Default value

No default value defined.

Campaign | partitions | partition[n] | audienceLevels | audienceLevel | field[n]

Properties in the this category define an audience level field. These properties are populated when a user creates audience levels on the Administration page in Campaign You should not edit properties in this category.

type

Description

The `partition[n] > audienceLevels > audienceLevel > field[n] > type` property is populated when a user creates audience levels on the Administration page in Campaign. You should not edit this property.

Default value

No default value defined.

name

Description

The `partition[n] > audienceLevels > audienceLevel > field[n] > name` property is populated when a user creates audience levels on the Administration page in Campaign. You should not edit this property.

Default value

No default value defined.

Campaign | partitions | partition[n] | dataSources

The properties in `Campaign|partitions|partition[n]|dataSources` determine how IBM Campaign interacts with databases, including its own system tables, for the specified partition.

These properties specify the databases that IBM Campaign can access and they control many aspects of how queries are formed.

Each data source that you add in IBM Campaign is represented by a category under `Campaign|partitions|partition[n]|dataSources|<data-source-name>`.

Note: The IBM Campaign system tables data source for each partition must be named `UA_SYSTEM_TABLES` in IBM Marketing Platform, and every IBM Campaign partition must have a **dataSources | UA_SYSTEM_TABLES** category on the Configuration page.

New category name

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Use the New category name field when you create a data source by clicking on one of the provided templates. The provided templates are indicated by italics and parentheses, for example *(DB2 Template)*. Enter a category name to identify the data source, such as DB2_Customers. After you save a new category, it appears in the navigation tree. You can change its properties as needed. The properties that are available depend on which template you selected. All of the possible properties for all templates are listed below in alphabetical order.

AccessLibrary

Description

IBM Campaign chooses its data source access library according to the data source type. For example, libora4d.so is used for Oracle connectivity, while libdb24d.so is used for DB2 connectivity. In most cases, the default selections are appropriate. However, the AccessLibrary property can be changed if the default value proves to be incorrect in your IBM Campaign environment. For example, 64-bit IBM Campaign provides two ODBC access libraries: one appropriate for ODBC data sources compatible with the unixODBC implementation (libodb4d.so) and the other compatible with the DataDirect implementation (libodb4dDD.so, used by IBM Campaign to access, for example, Teradata).

AliasPrefix

Description

The AliasPrefix property specifies the way IBM Campaign forms the alias name that IBM Campaign creates automatically when using a dimension table and writing to a new table.

Note that each database has a maximum identifier length; check the documentation for the database you are using to be sure that the value you set does not exceed the maximum identifier length for your database.

Default value

A

Additional libraries for AIX

Description

IBM Campaign includes two additional libraries for AIX ODBC driver managers that support the ODBC ANSI API rather than the ODBC Unicode API:

- libodb4dAO.so (32- and 64-bit): ANSI-only library for unixODBC-compatible implementations
- libodb4dDDAO.so (64-bit only): ANSI-only library for DataDirect-compatible implementations

If you determine that the default access library must be overridden, set this parameter (for example, to libodb4dDD.so, overriding the default selection of libodb4d.so).

Default value

No default value defined.

AllowBaseJoinsInSelect

Description

This property determines whether IBM Campaign attempts to do a SQL join of base tables (from the same data source) used in a Select process; otherwise, the equivalent join is done on the Campaign server.

Default value

TRUE

Valid Values

TRUE | FALSE

AllowSegmentUsingSQLCase

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property specifies whether the Segment process consolidates multiple SQL statements into a single SQL statement, when specific configuration conditions are met.

Setting this property to TRUE results in significant performance improvements when all of the following conditions are met:

- Segments are mutually exclusive.
- All segments come from a single table.
- Criteria for each segment are based on the IBM macro language.

In this case, IBM Campaign generates a single SQL CASE statement to perform segmentation, followed by segment-by-field processing on the Campaign application server.

Default value

TRUE

Valid Values

TRUE | FALSE

AllowTempTables

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property specifies whether IBM Campaign creates temporary tables in the database. Creating temporary tables can significantly improve the performance of campaigns.

When the value is TRUE, temporary tables are enabled. Each time a query is issued against the database (for example, by the Segment process), the resulting IDs are written to a temporary table in the database. When an additional query is issued, IBM Campaign can use that temporary table to retrieve rows from the database.

A number of IBM Campaign operations, such as `useInDbOptimization`, rely on the ability to create temp tables. If temporary tables are not enabled, IBM Campaign retains the selected IDs in the IBM Campaign server memory. The additional query retrieves IDs from the database and matches them to the IDs in server memory. This can negatively impact performance.

You must have appropriate privileges to write in the database to use temporary tables. Privileges are determined by the database login that you provide when you connect to the database.

Default value

`TRUE`

Note: Typically, you set **AllowTempTables** to `TRUE`. To override the value for a specific flowchart, open the flowchart in Edit mode, select **Admin** > **Advanced settings**, click the **Server optimization** tab, and select **Disallow use of temp tables for this flowchart**.

ASMSaveDBAuthentication

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

The `ASMSaveDBAuthentication` property specifies whether, when you log in to Campaign and map a table in a data source you did not previously log in to, IBM Campaign saves your user name and password in IBM Marketing Software.

If you set this property to `TRUE`, Campaign does not prompt you for a user name and password when you log in to the data source. If you set this property to `FALSE`, Campaign prompts you for a user name and password each time you log in to the data source.

Default value

`TRUE`

Valid Values

TRUE | FALSE

ASMUserForDBCredentials

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

The `ASMUserForDBCredentials` property specifies the IBM Marketing Software user name that is assigned to the IBM Campaign system user (required to access the Campaign system tables).

This property must be the same user that was created as the Campaign system user during installation. This property is undefined by default.

Default value

No default value defined.

BulkInsertBlockSize

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property defines the maximum size of a data block, in number of records, that Campaign passes to the database at a time.

Default value

100

BulkInsertRequiresColumnType

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The BulkInsertRequiresColumnType property is required to support DataDirect ODBC data sources only. Set this property to TRUE for DataDirect ODBC data sources when you use bulk (array) inserts. Set the property to FALSE to be compatible with most other ODBC drivers.

Default value

FALSE

BulkReaderBlockSize

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The BulkReaderBlockSize property defines the size of a data block, in number of records, that Campaign reads from the database at a time.

Default value

2500

ConditionalSQLCloseBracket

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The ConditionalSQLCloseBracket property specifies the type of bracket that is used to indicate the end of a conditional segment in raw SQL custom macros. Conditionalized segments that are enclosed in the specified open and close bracket type are used only if temp tables exist. They are ignored if there are no temp tables.

Default value

} (closing curly brace)

ConditionalSQLOpenBracket

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The ConditionalSQLOpenBracket property specifies the type of bracket used to indicate the start of a conditional segment in raw SQL custom macros. Conditionalized segments enclosed within the brackets specified by the ConditionalSQLOpenBracket and ConditionalSQLCloseBracket properties are used only if temp tables exist, and are ignored if there are no temp tables.

Default value

{ (opening curly brace)

ConnectionCacheSize

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The ConnectionCacheSize property specifies the number of connections that Campaign maintains in a cache for each data source.

By default (N=0), Campaign establishes a new connection to a data source for each operation; if Campaign maintains a cache of connections and a connection is available for reuse, Campaign uses the cached connection rather than establishing a new connection.

If the setting is not 0, when a process is done with a connection, Campaign keeps up to the specified number of connections open for an amount of time that is specified by the InactiveConnectionTimeout property. After this time expires, the connections are removed from the cache and closed.

Default value

0 (zero)

DateFormat

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Campaign uses the value of the DateFormat property to determine how to parse data in date formats when using the Campaign macro language or when interpreting data from date columns.

Set the value of the DateFormat property to the format in which Campaign expects to receive dates from this data source. The value must match the format that your database uses to display dates on select. For most databases, this setting is the same as the setting for the DateOutputFormatString property.

Note: If you use the multi-locale feature, do not use date formats that contain 3-letter months (MMM), %b (abbreviated month name), or %B (full month name). Instead, use a delimited or fixed format with a numeric value for the month.

To determine the date format that your database uses, select a date from the database. For additional information, see the following table.

Table 79. Date formats

Database	To determine the correct setting
DB2	Connect to the database from a machine that is running the Campaign server. Use `db2test` in the `Campaign\bin` directory to connect and issue the following command: `values current date` If your operating system does not provide the `db2test` utility, use the `cxntest` utility to test connections to the target database.
Hive-based Hadoop big data	All Date strings (Date, DateFormat, DateTimeFormat, DateTimeOutputFormatString) must use the dash "-" character to format dates. Hive does not support any other characters for dates. Example: %Y-%m-%d %H:%M:%S

Netezza®	Connect to the database from a machine that is running the Campaign server. Use `odbctest`, in the `Campaign\bin` directory, to connect and issue the following command:
	`CREATE TABLE date_test (f1 DATE);`
	`INSERT INTO date_test values (current_date);`
	`SELECT f1 FROM date_test;`
	
	Another way to select date format is to run following command:
	`SELECT current_date FROM ANY_TABLE limit 1;`
	
	where *ANY_TABLE* is the name of any existing table.

Oracle	Log in to the database from the machine that is running the Campaign server. Use SQL *Plus to connect and issue the following command: `SELECT sysdate FROM dual` The current date is returned in NLS_DATE_FORMAT for that client.
SQL Server	Connect to the database from a machine that is running the Campaign listener. Use `odbctest`, in the `Campaign\bin` directory, to connect and issue the following command: `SELECT getdate()` If the **Use regional settings when outputting currency, numbers, dates, and times** option is not checked in the ODBC data source configuration, then you cannot reset the date format. In general, it is easier to leave this setting cleared so that the date format configuration does not change for each language.
Teradata	Teradata allows you to define the date format on a per-column basis. In addition to `dateFormat` and `dateOutputFormatString`, you must set `SuffixOnCreateDateField`. To be consistent with the system table settings, use: • SuffixOnCreateDateField = `FORMAT 'YYYY-MM-DD'` • DateFormat = `DELIM_Y_M_D` • DateOutputFormatString = `%Y-%m-%d`

Default value

`DELIM_Y_M_D`

Valid Values

Any of the formats that are specified in the DATE macro

DateOutputFormatString

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The DateOutputFormatString property specifies the format of the date datatype to be used when Campaign writes any date, such as a campaign start or end date, to a database. Set the value of the DateOutputFormatString property to the format that the data source expects for columns of the type date. For most databases, this setting is the same as the setting for the [data_source_name] > DateFormat property.

The DateOutputFormatString property can be set to any of the formats that are specified for format_strin the DATE_FORMAT macro. The DATE_FORMAT macro accepts two different kinds of formats. One is an identifier (for example, DELIM_M_D_Y, DDMMMYYYY, the same as accepted by the DATE macro), while the other is a format string. The value of the DateOutputFormatString property must be a format string - it must not be one of the DATE macro identifiers. Typically, use one of the delimited formats.

You can verify whether you selected the correct format by creating a table and inserting a date in the format you selected, as described in the following procedure.

To verify DateOutputFormatString

1. Connect to the database using the appropriate tool, as described in the table for "Selecting a date by database".

   Do not use the query tools that come with the database (such as SQL Server's Query Analyzer) to verify that dates are being sent to the database correctly. These query tools might convert the date format to something other than what Campaign actually sent to the database.
2. Create a table and insert a date in the format you selected. For example, if you selected %m/%d/%Y:

   ```
   CREATE TABLE date_test (F1 DATE)
   INSERT INTO date_test VALUES ('03/31/2004')
   ```

   If the database allows the INSERT command to complete successfully, then you selected the correct format.

Default value

%Y/%m/%d

DateTimeFormat

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The value of the <data-source-name> DateTimeFormat property specifies the format in which Campaign expects to receive datetime/timestamp data from a database. It must match the format that your database uses to display datetime/timestamp data on select. For most databases, this setting is the same as the setting for DateTimeOutputFormatString.

Typically, you set the DateTimeFormat by prepending your DateFormat value with DT_ after determining the DateFormat value as described previously.

Note: If you use the multi-locale feature, do not use date formats that contain 3-letter months (MMM), %b (abbreviated month name), or %B (full month name). Instead, use a delimited or fixed format with a numeric value for the month.

Default value

DT_DELIM_Y_M_D

Valid Values

Only delimited formats are supported, as follows:

- DT_DELIM_M_D
- DT_DELIM_M_D_Y
- DT_DELIM_Y_M
- DT_DELIM_Y_M_D
- DT_DELIM_M_Y
- DT_DELIM_D_M
- DT_DELIM_D_M_Y

DateTimeOutputFormatString

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The DateTimeOutputFormatString property specifies the format of the datetime datatype to be used when Campaign writes any datetime, such as a campaign start or end date and time, to a database. Set the value of the DateTimeOutputFormatString property to the format that the data source expects for columns of the type datetime. For most databases, this setting is the same as the setting for the [data_source_name] > DateTimeFormat property.

See DateOutputFormatString for a method for verifying that the format you select is correct.

Default value

%Y/%m/%d %H:%M:%S

DB2NotLoggedInitially

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property determines whether IBM Campaign uses the not logged initially SQL syntax when populating temporary tables in DB2.

A value of TRUE disables logging for inserts into temp tables, which improves performance and decreases database resource consumption. When set to TRUE, if a temp table transaction fails for any reason, the table will become corrupted and must be dropped. All data previously contained in the table will be lost.

If your version of DB2 does not support the `not logged initially` syntax, set this property to `FALSE`.

If you are using a DB2 11 user database on z/OS®, set this property to `FALSE`. If you are using DB2 10.5 with the BLU feature ON for a user database, set both **DB2NotLoggedInitially** and **DB2NotLoggedInitiallyUserTables** to `FALSE`.

Default value

`TRUE`

Valid Values

`TRUE | FALSE`

DB2NotLoggedInitiallyUserTables

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

The `DB2NotLoggedInitiallyUserTables` property determines whether IBM Campaign uses the `not logged initially` SQL syntax for inserts into DB2 user tables.

A value of `TRUE` disables logging for inserts into the user tables, which improves performance and decreases database resource consumption. When set to `TRUE`, if a user table transaction fails for any reason, the table will become corrupted and must be dropped. All data previously contained in the table will be lost.

If you are using DB2 10.5 with the BLU feature ON for a user database, set both **DB2NotLoggedInitially** and **DB2NotLoggedInitiallyUserTables** to `FALSE`.

Note: The **DB2NotLoggedInitiallyUserTables** property is not used for the IBM Campaign system tables.

Default value

`FALSE`

Valid Values

`TRUE | FALSE`

DefaultScale

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

The `DefaultScale` property is used when Campaign creates a database field to store numeric values from a flat file or a derived field, when using the Snapshot or Export process.

This property is not used for numeric values originating in a database table, unless the database field omits information about precision and scale. (Precision indicates the total number of digits allowed for the field. Scale indicates the number of digits allowed to the right of the decimal point. For example, 6.789 has a precision of 4 and a scale of 3. Values obtained

from a database table include information about precision and scale, which Campaign uses when creating the field.)

Example: Flat files do not indicate precision and scale so you can use DefaultScale to specify how many places to the right of the decimal point to define for the field that is created, as shown below:

- DefaultScale=0 creates a field with no places to the right of the decimal point (only whole numbers can be stored).
- DefaultScale=5 creates a field with a maximum of 5 values to the right of the decimal point.

If the value set for DefaultScale exceeds the field's precision, DefaultScale=0 is used for those fields. For example, if the precision is 5, and DefaultScale=6, a value of zero is used.

Default value

0 (zero)

DefaultTextType

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The DefaultTextType property is intended for ODBC data sources. This property tells Campaign how to create text fields in the destination data source if the source text fields are from a different data source type. For example, the source text fields might be from a flat file or from a different type of DBMS. If the source text fields are from the same type of DBMS, this property is ignored and the text fields are created in the destination data source using the data types from the source text fields.

Default value

VARCHAR

Valid Values

VARCHAR | NVARCHAR

DeleteAsRecreate

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The DeleteAsRecreate property specifies whether, when an output process is configured to REPLACE TABLE and if TRUNCATE is not supported, Campaign drops and recreates the table or only deletes from the table.

When the value is TRUE, Campaign drops the table and recreates it.

When the value is FALSE, Campaign executes a DELETE FROM from the table.

Default value

FALSE

Valid Values

TRUE | FALSE

DeleteAsTruncate

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The DeleteAsTruncate property specifies whether, when an output process is configured to REPLACE TABLE, Campaign uses TRUNCATE TABLE or deletes from the table.

When the value is TRUE, Campaign runs a TRUNCATE TABLE from the table.

When the value is FALSE, Campaign runs a DELETE FROM from the table.

The default value depends on the database type.

Default value

- TRUE for Netezza, Oracle, and SQLServer.
- FALSE for other database types.

Valid Values

TRUE | FALSE

DisallowTempTableDirectCreate

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property is used by Oracle, Netezza, and SQL Server data sources and is ignored for all other data sources.

This property specifies the way Campaign adds data to a temp table.

When set to FALSE, Campaign performs direct create-and-populate SQL syntax using one command. For example: CREATE TABLE <table_name> AS ... (for Oracle and Netezza) and SELECT <field_names> INTO <table_name> ... (for SQL Server).

When set to TRUE, Campaign creates the temp table and then populates it directly from table to table using separate commands.

Default value

FALSE

Valid Values

TRUE | FALSE

DSN

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Set this property to the data source name (DSN) as assigned in your ODBC configuration for this Campaign data source. For SQL server, set this property to the DSN (data source name) that was created during installation. For Oracle and DB2, set this property to the database name or the SID (service) name. This value is undefined by default.

Using the Campaign data source configuration properties, you can specify multiple logical data sources that refer to the same physical data source. For example, you can create two sets of data source properties for the same data source, one with `AllowTempTables = TRUE` and the other with `AllowTempTables = FALSE`. Each of these data sources has a different name in Campaign, but if they refer to the same physical data source they have the same DSN value.

Default value

No default value defined.

DSNUsingOSAuthentication

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

The `DSNUsingOSAuthentication` property applies only when an Campaign data source is SQL Server. Set the value to `TRUE` when the DSN is configured to use Windows Authentication mode.

Default value

`FALSE`

Valid Values

`TRUE | FALSE`

EnableBaseDimSelfJoin

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

The `EnableBaseDimSelfJoin` property specifies whether the Campaign database behavior will perform self-joins when the Base and Dimension tables are mapped to the same physical table and the Dimension is not related to the Base table on the Base table's ID field(s).

By default, this property is set to `FALSE`, and when the Base and Dimension tables are the same database table and the relationship fields are the same (for example, AcctID to AcctID), Campaign assumes that you do not want to perform a join.

Default value

`FALSE`

EnableSelectDistinct

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

The `EnableSelectDistinct` property specifies whether the internal lists of IDs for Campaign are de-duplicated by the Campaign server or by the database.

When the value is TRUE, the database performs de-duplication, and SQL queries generated against the database then have the form (when appropriate):

SELECT DISTINCT *key* FROM *table*

When the value is FALSE, the Campaign server performs de-duplication, and SQL queries generated against the database have the form:

SELECT *key* FROM *table*

Leave the default value of FALSE if:

- Your database is constructed so that unique identifiers (primary keys of base tables) are already guaranteed to be de-duped.
- You want the Campaign application server to perform de-duplication to reduce resource consumption/burden on the database.

Regardless of what value you specify for this property, Campaign automatically ensures that keys are de-duplicated as required. This property merely controls where the de-duplication effort occurs (on the database or on the Campaign server).

Default value

TRUE

Valid Values

TRUE | FALSE

EnableSelectOrderBy

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The EnableSelectOrderBy property specifies whether the internal lists of IDs for Campaignare sorted by the Campaign server or by the database.

When the value is TRUE, the database performs the sorting, and SQL queries generated against the database have the form:

SELECT <*key*> FROM <*table*> ORDER BY <*key*>

When the value is FALSE, the Campaign server performs the sorting, and SQL queries generated against the database have the form:

SELECT <*key*> FROM <*table*>

Note: Only set this property to FALSE if the audience levels used are text strings on a non-English database. All other scenarios can use the default of TRUE.

Default value

TRUE

Valid Values

True | False

ExcludeFromTableDisplay

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The ExcludeFromTableDisplay parameter allows you to limit the database tables that are displayed during table mapping in IBM Campaign. It does not reduce the number of table names retrieved from the database. Table names that match the specified patterns are not displayed. Values for this parameter are case-sensitive.

Example: If you set the value to sys.*, tables with names that begin with all lower case sys. are not displayed.

Example: UAC_* (the default value for SQL Server data sources) excludes temp tables and Extract tables, when the ExtractTablePrefix property's value is the default value.

Example: To exclude the IBM Marketing Platform system tables, as they are not relevant when working with user data:

DF_*,USM_*,OLS_*,QRTZ*,USCH_*,UAR_*

Using Oracle as an example, the complete value would be:

UAC_*,PUBLIC.*,SYS.*,SYSTEM.*,DF_*,USM_*,OLS_*,QRTZ*, USCH_*,UAR_*

Default value

UAC_*,PUBLIC.*,SYS.*,SYSTEM.* (for an Oracle data source)

UAC_* (for a SQL Server data source)

UAC_*,SYSCAT.*,SYSIBM.*,SYSSTAT.* (for a DB2 data source)

ExtractTablePostExecutionSQL

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Use the ExtractTablePostExecutionSQL property to specify one or more complete SQL statements that run immediately after the creation and population of an Extract table.

Tokens available to ExtractTablePostExecutionSQL are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which the Extract table was created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which the Extract table was created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which the Extract table was created.
<DBUSER>	This token is replaced with the database user name for the database where the Extract table was created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the Extract table creation.

Token	Description
<KEYCOLUMNS>	This token is replaced with the Extract table column name(s).
<TABLENAME>	This token is replaced with the Extract table name.
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

Not defined

Valid Values

A valid SQL statement

ExtractTablePrefix

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

The `ExtractTablePrefix` property specifies a string that is automatically prepended to all Extract table names in Campaign. This property is useful when two or more data sources point to the same database. For details, see the `TempTablePrefix` description.

Default value

`UAC_EX`

ForceNumeric

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

The `ForceNumeric` property specifies whether Campaign retrieves numeric values as the data type `double`. When the value is set to `TRUE`, Campaign retrieves all numeric values as the data type `double`.

Default value

`FALSE`

Valid Values

`TRUE | FALSE`

HiveQueryMode

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

This property is used only for Hive-based Hadoop data sources (BigDataODBCHiveTemplate). It provides a way to switch between the DataDirect and Cloudera drivers. For DataDirect, select `Native`. For Cloudera, select `SQL`.

Valid Values

Native | SQL

InactiveConnectionTimeout

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The InactiveConnectionTimeout property specifies the number of seconds an inactive Campaign database connection is left open before it is closed. Setting the value to 0 disables the timeout, leaving the connection open.

Default value

120

InsertLogSize

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The InsertLogSize property specifies when a new entry is entered in the log file while the Campaign Snapshot process is running. Every time the number of records written by the Snapshot process reaches a multiple of the number specified in the InsertLogSize property, a log entry is written. The log entries can help you determine how far a running Snapshot process has progressed. Setting this value too low may create large log files.

Default value

100000 (one hundred thousand records)

Valid Values

Positive integers

JndiName

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The JndiName property is used only when configuring the Campaign system tables (not for user data sources). Set its value to the Java Naming and Directory Interface (JNDI) data source that you created in your application server (WebSphere or WebLogic) to connect to this data source.

Default value

campaignPartition1DS

LoaderCommand

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property specifies the command issued to invoke your database load utility in IBM Campaign. If you set this property, IBM Campaign enters the database loader utility mode for all output files from the Snapshot process

that are used with the **Replace all records** settings. This property also invokes the database loader utility mode when IBM Campaign uploads ID lists into temp tables.

The valid value for this property is any full path name either to the database load utility executable or to a script that launches the database load utility. Using a script allows you to perform additional setup before invoking the load utility.

Note: If you use IBM Contact Optimization and you are configuring loader settings for the UA_SYSTEM_TABLES datasource, there are important considerations to take into account. For example, you must use absolute paths for **LoaderCommand** and **LoaderCommandForAppend**. Read about setting up Campaign to use database load utilities in the *IBM Campaign Administrator's Guide*.

Most database load utilities require several arguments to be launched successfully. These arguments can include specifying the data file and control file to load from and the database and table to load into. IBM Campaign supports the following tokens, which are replaced by the specified elements when the command is run. Consult your database load utility documentation for the correct syntax to use when invoking your database load utility.

This property is undefined by default.

Tokens available to **LoaderCommand** are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart being run.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart being run.
<CONTROLFILE>	This token is replaced with the full path and file name to the temporary control file that IBM Campaign generates according to the template that is specified in the **LoaderControlFileTemplate** property.
<DATABASE>	This token is replaced with the name of the data source that IBM Campaign is loading data into. This is the same data source name used in the category name for this data source.
<DATAFILE>	This token is replaced with the full path and file name to the temporary data file created by IBM Campaign during the loading process. This file is in the IBM Campaign Temp directory, `UNICA_ACTMPDIR`.
<DBUSER>	This token is replaced with the database user name for the database.

Token	Description
<DSN>	This token is replaced with the value of the DSN property. If the DSN property is not set, the <DSN> token is replaced by the data source name used in the category name for this data source (the same value used to replace the <DATABASE> token).
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart being run.
<NUMFIELDS>	This token is replaced with the number of fields in the table.
<PASSWORD>	This token is replaced with the database password from the current flowchart connection to the data source.
<TABLE>	This token is obsolete. Use <TABLENAME> instead.
<TABLENAME>	This token is replaced with the database table name that IBM Campaign is loading data into. This is the target table from your Snapshot process or the name of the Temp Table being created by IBM Campaign.
<USER>	This token is replaced with the database user from the current flowchart connection to the data source.

Default value

No default value defined.

Valid Values

Any full path name either to the database load utility executable or to a script that launches the database load utility.

LoaderCommandForAppend

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property specifies the command issued to invoke your database load utility for appending records to a database table in IBM Campaign. If you set this property, IBM Campaign enters database loader utility mode for all output files from the Snapshot process that are used with the **Append Records** settings.

This property is specified as a full path name either to the database load utility executable or to a script that launches the database load utility. Using a script allows you to perform additional setup before invoking the load utility.

Most database load utilities require several arguments to be successfully launched. These can include specifying the data file and control file to load from and the database and table to load into. The tokens are replaced by the specified elements when the command is run.

Consult your database load utility documentation for the correct syntax to use when invoking your database load utility.

This property is undefined by default.

Tokens available to **LoaderCommandForAppend** are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart being run.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart being run.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart being run.
<CONTROLFILE>	This token is replaced with the full path and file name to the temporary control file that Campaign generates according to the template that is specified in the **LoaderControlFileTemplate** property.
<DATABASE>	This token is replaced with the name of the data source that IBM Campaign is loading data into. This is the same data source name used in the category name for this data source.
<DATAFILE>	This token is replaced with the full path and file name to the temporary data file created by IBM Campaign during the loading process. This file is in the Campaign Temp directory, `UNICA_ACTMPDIR`.
<DBUSER>	This token is replaced with the database user name for the database where the temp tables were created.
<DSN>	This token is replaced with the value of the DSN property. If the `DSN` property is not set, the `<DSN>` token is replaced by the data source name used in the category name for this data source (the same value used to replace the `<DATABASE>` token).
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the temp table creation.
<NUMFIELDS>	This token is replaced with the number of fields in the table.
<PASSWORD>	This token is replaced with the database password from the current flowchart connection to the data source.
<TABLE>	This token is obsolete. Use `<TABLENAME>` instead.
<TABLENAME>	This token is replaced with the database table name that IBM Campaign is loading data into. This is the target table from your Snapshot process or the name of the Temp Table being created by IBM Campaign.

Token	Description
<USER>	This token is replaced with the database user from the current flowchart connection to the data source.

Default value

No default value defined.

LoaderControlFileTemplate

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property specifies the full path and file name to the control file template that is configured for IBM Campaign. The path to the template is relative to the current partition. For example: loadscript.db2.

When this property is set, IBM Campaign dynamically builds a temporary control file based on the specified template. The path and name of this temporary control file is available to the <CONTROLFILE> token that is available to the **LoaderCommand** property.

Before you use IBM Campaign in the database loader utility mode, you must configure the control file template that is specified by this parameter. The control file template supports the following tokens, which are dynamically replaced when the temporary control file is created by IBM Campaign.

For the correct syntax required for your control file, see your database loader utility documentation.

This property is undefined by default.

Tokens available to **LoaderControlFileTemplate** are the same as those described for the **LoaderCommand** property, plus the following special tokens, which are repeated once for each field in the outbound table.

Token	Description
<DBCOLUMNNUMBER>	This token is replaced with the column ordinal in the database.
<FIELDLENGTH>	This token is replaced with the length of the field being loaded into the database.
<FIELDNAME>	This token is replaced with the name of the field being loaded into the database.
<FIELDNUMBER>	This token is replaced with the number of the field being loaded into the database.
<FIELDTYPE>	This token is replaced with the literal CHAR( ). The length of this field is specified between the parentheses (). If your database does not understand the field type CHAR, you can manually specify the appropriate text for the field type and use the <FIELDLENGTH> token. For example, for SQLSVR and SQL2000 you would use SQLCHAR(<FIELDLENGTH>).
<NATIVETYPE>	This token is replaced with the actual database type that this field is loaded into.

Token	Description
<xyz>	This token places the specified character(s) on all fields being loaded into the database, except the last. A typical use is <,> which repeats a comma for all fields except the last.
<~xyz>	This token places the specified characters only on the last repeated line.
<!xyz>	This token places the specified character(s), including the angle brackets < >, on all lines.

Default value

No default value defined.

LoaderControlFileTemplateForAppend

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property specifies the full path and file name to the control file template that is configured in IBM Campaign. The path to the template is relative to the current partition. For example: loadappend.db2

When this property is set, IBM Campaign dynamically builds a temporary control file based on the specified template. The path and name of this temporary control file is available to the <CONTROLFILE> token that is available to the **LoaderCommandForAppend** property.

Before you use IBM Campaign in the database loader utility mode, you must configure the control file template that is specified by this property. See your database loader utility documentation for the correct syntax required for your control file.

The available tokens are the same as the tokens for the **LoaderControlFileTemplate** property.

This property is undefined by default.

Default value

No default value defined.

LoaderDelimiter

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property specifies whether the temporary data file is a fixed-width or delimited flat file, and, if it is delimited, the characters that IBM Campaign uses as delimiters.

If the value is undefined, IBM Campaign creates the temporary data file as a fixed width flat file.

If you specify a value, it is used when the loader is invoked to populate a table that is known to be empty. IBM Campaign creates the temporary data file as a delimited flat file, using the value of this property as the delimiter.

The delimiter is a character such as comma (,) or semi-colon (;) that separates fields in the temporary data files that are loaded into the user data source.

Important: The following fields, if used, must use the same character as specified for **LoaderDelimiter**: **SuffixOnTempTableCreation**, **SuffixOnSegmentTableCreation**, **SuffixOnSnapshotTableCreation**, **SuffixOnExtractTableCreation**, **SuffixOnUserBaseTableCreation**, **SuffixOnUserTableCreation**.

Important: For big data, such as Hadoop Hive or Amazon Redshift, the delimiter value must match the ROW format delimiter that was used when the big data database table was created. In this example, a comma is used: **ROW FORMAT DELIMITED FIELDS TERMINATED BY** ',' ;"

This property is undefined by default.

Default value

No default value defined.

Valid Values

Characters, which can be enclosed in double quotation marks, if wanted. Hive-based Hadoop big data does not support the Tab (/t) character.

LoaderDelimiterAtEnd

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

Some external load utilities require that the data file be delimited and that each line end with the delimiter. To accommodate this requirement, set the **LoaderDelimiterAtEnd** value to `TRUE`, so that when the loader is invoked to populate a table that is known to be empty, IBM Campaign uses delimiters at the end of each line. For example, DB2 on Unix expects each record to be terminated by a line feed character only; Campaign Campaign on Windows uses carriage return and line feed characters. Putting a delimiter at the end of every record ensures that the last column in the data file will load properly.

`FALSE`

Default value

`FALSE`

Valid Values

`TRUE | FALSE`

LoaderDelimiterAtEndForAppend

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

Some external load utilities require that the data file be delimited and that each line end with the delimiter. To accommodate this requirement, set the **LoaderDelimiterAtEndForAppend** value to `TRUE`, so that when the loader is invoked to populate a table that is not known to be empty, IBM

Campaign uses delimiters at the end of each line. For example, DB2 on Unix expects each record to be terminated by a line feed character only; IBM Campaign on Windows uses carriage return and line feed characters. Putting a delimiter at the end of every record ensures that the last column in the data file will load properly.

Default value

FALSE

Valid Values

TRUE | FALSE

LoaderDelimiterForAppend

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property specifies whether the temporary IBM Campaign data file is a fixed-width or delimited flat file, and, if it is delimited, the character or set of characters used as delimiters.

If the value is undefined, IBM Campaign creates the temporary data file as a fixed width flat file.

If you specify a value, it is used when the loader is invoked to populate a table that is not known to be empty. IBM Campaign creates the temporary data file as a delimited flat file, using the value of this property as the delimiter.

This property is undefined by default.

Default value

No default value defined.

Valid Values

Characters, which you may enclose in double quotation marks, if wanted.

LoaderPostLoadDataFileRemoveCmd

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property is used only for Hive-based Hadoop data sources (BigDataODBCHiveTemplate). This property is used together with LoaderPreLoadDataFileCopyCmd. After data files are copied from Campaign to the /tmp folder on the Hive-based Hadoop system, the LoaderPostLoadDataFileRemoveCmd property uses the SSH "rm" command to remove the temporary data file.

For example: ssh mapr@example.com "rm/tmp/<DATAFILE>"

For important information, read about exporting data from Campaign to a Hive-based Hadoop system.

Default value

none

LoaderPreLoadDataFileCopyCmd

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property is used only for Hive-based Hadoop data sources (BigDataODBCHiveTemplate). This property uses SCP to copy data from IBM Campaign to a temp folder called /tmp on your Hive-based Hadoop system. The location must be called /tmp and it must be on the Hive server (the file system location, not the HDFS location). You can either specify an SCP command or call a script that specifies the SCP command.

Example #1: scp <DATAFILE> mapr@example.com:/tmp

Example #2: /opt/IBM/CampaignBigData/bin/copyToHive.sh <DATAFILE>

In addition to this property, use LoaderPostLoadDataFileRemove to remove the temporary data file from the Hive server after it has been copied.

For important information, read about exporting data from Campaign to a Hive-based Hadoop system.

Default value

none

LoaderNULLValueInDelimitedData

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property supports null values in delimited data for database loaders, specifically Netezza. Enter the string that represents a null value for the column.

Default value

null

LoaderUseLocaleDP

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property specifies, when IBM Campaign writes numeric values to files to be loaded by a database load utility, whether the locale-specific symbol is used for the decimal point.

Set this value to FALSE to specify that the period (.) is used as the decimal point.

Set this value to TRUE to specify that the decimal point symbol appropriate to the locale is used.

Default value

FALSE

Valid Values

TRUE | FALSE

MaxItemsInList

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Allows you to specify the maximum number of items that IBM Campaign is allowed to include in a single list in SQL (for example, the list of values following an IN operator in a WHERE clause).

Default value

1000 (Oracle only), 0 (unlimited) for all other databases

Valid Values

integers

MaxQueryThreads

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property specifies the upper limit on the number of simultaneous queries allowed to run against each database source from a single IBM Campaign flowchart. Higher values generally improve performance.

IBM Campaign runs database queries using independent threads. Because IBM Campaign processes run in parallel, it is common to have multiple queries running simultaneously against a single data source. If the number of queries to be run in parallel exceeds the MaxQueryThreads, the IBM Campaign server limits the number of simultaneous queries to the specified value.

The maximum value is unlimited.

Note: If maxReuseThreads is set to a non-zero value, it should be greater than or equal to the value of MaxQueryThreads.

Default value

Varies depending on the database

MaxRowFetchRecords

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

For performance reasons, it is best to keep this number low.

When the selected number of IDs is less than the value specified by the MaxRowFetchRecords property, IBM Campaign passes the IDs to the database one at a time, in separate SQL queries. This process can be very time-consuming. If the number of selected IDs is greater than the value specified by this property, IBM Campaign uses temporary tables (if allowed on the database source), or it pulls down all the values from the table, not including any unnecessary values.

Default value

100

MaxTempTableJoinPctSelectAll

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

When a query is issued, IBM Campaign creates a temporary table on the database containing the exact list of IDs, as a result of the query. When an additional query that selects all records is issued against the database, the MaxTempTableJoinPctSelectAll property specifies whether a join is performed with the temporary table.

If the relative size of the temporary table (specified as a percentage) is greater than the value of the MaxTempTableJoinPctSelectAll property, no join is performed. All records are selected first, then unwanted records are discarded.

If the relative size of the temporary table (specified as a percentage) is less than or equal to the value of MaxTempTableJoinPctSelectAll property, the join is performed with the temporary table first, and then the resulting IDs are retrieved to the server.

This property is applicable only if the value of the AllowTempTables property is set to TRUE. This property is ignored if the useInDbOptimization property is set to YES.

Default value

90

Valid Values

Integers between 0-100. A value of 0 means that temporary table joins are never used; a value of 100 means that table joins are always used, regardless of the size of the temporary table.

Example

Assume that MaxTempTableJoinPctSelectAll is set to 90. First, you might want to select customers (CustID) with account balances (Accnt_balance) greater than $1,000 from the database table (Customer).

The corresponding SQL expression generated by the Select process may look like this:

```
SELECT CustID FROM Customer
 WHERE Accnt_balance > 1000
```

The Select process may retrieve 100,000 IDs from the total table size of 1,000,000, which is 10%. If temporary tables are allowed, IBM Campaign writes the selected IDs (TempID) into a temporary table (Temp_table) in the database.

Then, you might want to snapshot the selected IDs (CustID) together with the actual balance (Accnt_balance). Since the relative size of the temporary table (Temp_table) is less than 90 percent (MaxTempTableJoinPctSelectAll), the join is done with the temporary table first. The SQL expression generated by the Snapshot process may look like this:

```
SELECT CustID, Accnt_balance FROM Customer, Temp_table WHERE CustID = TempID
```

If the Select process retrieves more than 90 percent, the subsequent Snapshot process retrieves all the records, and matches them with the first set of IDs, discarding the unnecessary ones.

The SQL expression generated by the Snapshot process may look like this:

```
SELECT CustID, Accnt_balance FROM Customer
```

MaxTempTableJoinPctWithCondition

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

When a query is issued, IBM Campaign creates a temporary table on the database containing the exact list of IDs, as a result of the query. When an additional query, selecting records with limitation conditions is issued against the database, the MaxTempTableJoinPctWithCondition property specifies whether a join should be performed with the temporary table.

If the relative size of the temporary table (specified as a percentage) is greater than the value of MaxTempTableJoinPctWithCondition, no join is performed. This avoids the overhead in the database where it may not be needed. In this case, the query is issued against the database, the resulting list of IDs retrieved, and then unwanted records are discarded as they are matched to the list in server memory.

If the relative size of the temporary table (in percentage) is less than or equal to the value of MaxTempTableJoinPctWithCondition, the join is done with the temporary table first, and then the resulting IDs are retrieved to the server.

This property is applicable only if the value of the AllowTempTables property is set to TRUE.

Default value

20

Valid Values

Integers between 0-100. A value of 0 means that temporary table joins are never used; a value of 100 means that table joins are always used, regardless of the size of the temporary table.

MinReqForLoaderCommand

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Use this property to set the threshold for using the bulk loader. IBM Campaign invokes the script assigned to the LoaderCommand propery when the number of unique IDs in the input cell exceeds the value defined here. The value of this property does not represent the number of records that will be written.

If this property is not configured, IBM Campaign assumes that the value is the default value (zero). If this property is configured but a negative value or non-integer value is set as the value, a value of zero is assumed.

Default value

0 (zero)

Valid Values

Integers

MinReqForLoaderCommandForAppend

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

Use this property to set the threshold for using the bulk loader. IBM Campaign invokes the script assigned to the `LoaderCommandForAppend` parameter when the number of unique IDs in the input cell exceeds the value defined here. The value of this property does not represent the number of records that will be written.

If this property is not configured, IBM Campaign assumes that the value is the default value (zero). If this property is configured but a negative value or non-integer value is set as the value, a value of zero is assumed.

Default value

`0 (zero)`

Valid Values

Positive integers

NumberOfRetries

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

The `NumberOfRetries` property specifies the number of times IBM Campaign automatically retries a database operation on failure. IBM Campaign automatically resubmits queries to the database this number of times before reporting a database error or failure.

Default value

`0 (zero)`

ODBCTableTypes

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

This property is empty by default, which is appropriate for all currently supported data sources.

Default value

Not defined

Valid Values

(empty)

ODBCUnicode

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The ODBCUnicode property specifies the type of encoding used in IBM Campaign ODBC calls. It is used only with ODBC data sources and is ignored when used with Oracle or DB2 native connectivity.

Important: If this property is set to UTF-8 or UCS-2, the data source's StringEncoding value must be set to either UTF-8 or WIDEUTF-8, otherwise the ODBCUnicode property's setting is ignored.

Default value

disabled

Valid Values

Possible values for this property are:

- Disabled: IBM Campaign uses ANSI ODBC calls.
- UTF-8: IBM Campaign uses Unicode ODBC calls and assumes that a SQLWCHAR is a single byte. This is compatible with DataDirect ODBC drivers.
- UCS-2: IBM Campaign uses Unicode ODBC calls and assumes that a SQLWCHAR is 2 bytes. This is compatible with Windows and unixODBC ODBC drivers.

ODBCv2

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Use the ODBCv2 property to specify which ODBC API specification IBM Campaign should use for the data source.

The default value of FALSE allows IBM Campaign to use the v3 API specification, while a setting of TRUE causes IBM Campaign to use the v2 API specification. Set the ODBCv2 property to TRUE for data sources that do not support the ODBC v3 API specification.

When the ODBCv2 property is set to TRUE, IBM Campaign does not support the ODBC Unicode API, and values other than disabled for the ODBCUnicode property are not recognized.

Default value

FALSE

Valid Values

TRUE | FALSE

OwnerForTableDisplay

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Use this property to limit the table mapping display in IBM Campaign to

tables in a specified schema. For example, to specify tables in the schema "dbo", set OwnerForTableDisplay=dbo.

Default value

No default value defined.

PadTextWithSpaces

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

When set to `TRUE`, the `PadTextWithSpaces` property causes IBM Campaign to pad text values with spaces until the string is the same width as the database field.

Default value

`FALSE`

Valid Values

`TRUE | FALSE`

PostExtractTableCreateRunScript

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

Use this property to specify a script or executable for IBM Campaign to run after an Extract table has been created and populated.

Tokens available to **PostExtractTableCreateRunScript** are described below.

Token	Description
<DBUSER>	This token is replaced with the database user name for the database where the Extract table was created.
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which the Extract table was created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which the Extract table was created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which the Extract table was created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the Extract table creation.
<PASSWORD>	This token is replaced with the database password from the current flowchart connection to the data source.
<KEYCOLUMNS>	This token is replaced with the Extract table column name(s).

Default value

Not defined

Valid Values

File name of a shell script or executable

PostSegmentTableCreateRunScript

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Specifies a script or executable that IBM Campaign runs after a Segment temp table has been created and populated.

Tokens available to **PostSegmentTableCreateRunScript** are described below.

Token	Description
<DBUSER>	This token is replaced with the database user name for the database where the Segment temp table was created.
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which the Segment temp table was created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which the Segment temp table was created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which the Segment temp table was created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the Segment temp table creation.
<PASSWORD>	This token is replaced with the database password from the current flowchart connection to the data source.
<KEYCOLUMNS>	This token is replaced with the Segment temp table column name(s).

Default value

Not defined

Valid Values

File name of a script or executable

PostSnapshotTableCreateRunScript

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Use the **PostSnapshotTableCreateRunScript** property to specify a script or executable that Campaign runs after a Snapshot table has been created and populated.The property is only called when snapshot process writes to "Extract Tables"

Tokens available to **PostSnapshotTableCreateRunScript** are described below.

Token	Description
<DBUSER>	This token is replaced with the database user name for the database where the Snapshot table was created.
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which the Snapshot table was created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which the Snapshot table was created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which the Snapshot table was created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the Snapshot table creation.
<PASSWORD>	This token is replaced with the database password from the current flowchart connection to the data source.
<KEYCOLUMNS>	This token is replaced with the Snapshot table column name(s).

Default value

Not defined

Valid Values

File name of a shell script or executable

PostTempTableCreateRunScript

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

Use the **PostTempTableCreateRunScript** property to specify a script or executable for Campaign to run after a temp table has been created and populated in a user data source or in the system tables database.

Tokens available to **PostTempTableCreateRunScript** are described below.

Token	Description
<DBUSER>	This token is replaced with the database user name for the database where the temp tables were created.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which temp tables were created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which temp tables were created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which temp tables were created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the temp table creation.
<PASSWORD>	This token is replaced with the database password from the current flowchart connection to the data source.
<KEYCOLUMNS>	This token is replaced with the temp table column name(s).

Default value

No default value defined.

PostUserTableCreateRunScript

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

Specifies a script or executable that Campaign runs after a User table has been created and populated.

Tokens available to `PostUserTableCreateRunScript` are described below.

Token	Description
<DBUSER>	This token is replaced with the database user name for the database where the User table was created.
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which the User table was created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which the User table was created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which the User table was created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the User table creation.
<PASSWORD>	This token is replaced with the database password from the current flowchart connection to the data source.

Token	Description
<KEYCOLUMNS>	This token is replaced with the User table column name(s).

Default value

Not defined

Valid Values

File name of a script or executable

PrefixOnSelectSQL

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

Use the `PrefixOnSelectSQL` property to specify a string that is automatically prepended to all `SELECT` SQL expressions generated by Campaign.

This property applies only to SQL generated by Campaign, and does not apply to SQL in raw SQL expressions used in the Select process.

This property is automatically added to the `SELECT` SQL expression without checking its syntax. If you use this property, make sure that it is a legal expression.

This property is undefined by default.

Tokens available to **PrefixOnSelectSQL** are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which temp tables were created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which temp tables were created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which temp tables were created.
<DBUSER>	This token is replaced with the database user name for the database where the temp tables were created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the temp table creation.
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

No default value defined.

QueryThreadSleep

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The QueryThreadSleep property affects the CPU utilization of the Campaign server process (UNICA_ACSVR). When the value is TRUE, the thread that the Campaign server process uses to check for query completion sleeps between checks. When the value is FALSE, the Campaign server process checks continuously for query completion.

Default value

TRUE

ReaderLogSize

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The ReaderLogSize parameter defines when Campaign makes a new entry in the log file when reading data from the database. Every time the number of records read from the database reaches a multiple of the number defined by this parameter, a log entry is written in the log file.

This parameter can help you determine how far a process has progressed in its run. Setting this value too low may create large log files.

Default value

1000000 (one million records)

Valid Values

Integers

SegmentTablePostExecutionSQL

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Use the SegmentTablePostExecutionSQL property to specify a complete SQL statement that Campaign runs after a Segment temp table has been created and populated.

Tokens available to **SegmentTablePostExecutionSQL** are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which the Segment temp table was created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which the Segment temp table was created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which the Segment temp table was created.

Token	Description
<DBUSER>	This token is replaced with the database user name for the database where the Segment temp table was created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the Segment temp table creation.
<KEYCOLUMNS>	This token is replaced with the Segment temp table column name(s).
<TABLENAME>	This token is replaced with the Segment temp table name.
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

Not defined

Valid Values

A valid SQL statement

SegmentTempTablePrefix

Description

Sets the prefix for Segment tables created by the CreateSeg process in this data source. This property is useful when two or more data sources point to the same database. For details, see the `TempTablePrefix` description.

Default value

UACS

SnapshotTablePostExecutionSQL

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

Use the `SnapshotTablePostExecutionSQL` property to specify one or more complete SQL statements to run immediately after a Snapshot table has been created and populated. This property is invoked only when a Snapshot process box writes out to an extract table.

Tokens available to **SnapshotTablePostExecutionSQL** are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which the Snapshot table was created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which the Snapshot table was created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which the Snapshot table was created.

Token	Description
<DBUSER>	This token is replaced with the database user name for the database where the Snapshot table was created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the Snapshot table creation.
<KEYCOLUMNS>	This token is replaced with the Snapshot table column name(s).
<TABLENAME>	This token is replaced with the Snapshot table name.
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

Not defined

Valid Values

A valid SQL statement

SQLOnConnect

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

The `SQLOnConnect` property defines a complete SQL statement that Campaign runs immediately after each database connection.

The SQL statement generated by this property is automatically passed to your database without checking its syntax. If you use this property, make sure that it is a legal expression. The string may be enclosed in quotation marks, but this is not required.

This property is undefined by default.

Tokens available to `SQLOnConnect` are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which temp tables were created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which temp tables were created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which temp tables were created.
<DBUSER>	This token is replaced with the database user name for the database where the temp tables were created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the temp table creation.

Token	Description
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

No default value defined.

StringEncoding

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The StringEncoding property specifies the character encoding of the database. When Campaign retrieves data from the database, the data is transcoded from the encoding specified to the internal encoding of Campaign (UTF-8). When Campaign sends a query to the database, character data is transcoded from the internal encoding of Campaign (UTF-8) to the encoding specified in the StringEncoding property.

The value of this property must match the encoding used on the database client.

Do not leave this value blank although it is undefined by default.

If you use ASCII data, set this value to UTF-8.

If your database client encoding is UTF-8, the preferred setting for this value is WIDEUTF-8. The WIDE-UTF-8 setting works only if your database client is set to UTF-8.

If you use the partitions > partition[n] > dataSources > data_source_name > ODBCUnicode property, set the StringEncoding property to either UTF-8 or WIDEUTF-8. Otherwise, the ODBCUnicode property value is ignored.

For a list of supported encodings, see *Character encodings in Campaign* in the *Campaign Administrator's Guide*.

Important: See the following sections for important exceptions and additional considerations.

Default value

No default value defined.

Database-specific considerations

This section describes how to set the correct values for DB2, SQL Server, or Teradata databases.

DB2

Identify the DB2 database code page and code set. For localized environments, the DB2 database must have the following configuration:

- Database code set = UTF-8
- Database code page = 1208

Set the StringEncoding property values in Campaign to the DB2 database code set value.

Set the DB2CODEPAGE DB2 environment variable to the DB2 database code page value:

- On Windows: Add the following line to the Campaign Listener startup script (<CAMPAIGN_HOME>\bin\cmpServer.bat):

  db2set DB2CODEPAGE=1208

- On UNIX: After DB2 is started, the system administrator must type the following command from the DB2 instance user:

  $ db2set DB2CODEPAGE=1208

  Then start the Campaign listener by running this command:

  ./rc.unica_ac start

This setting affects all DB2 data sources and can affect other running programs.

SQL Server

For SQL Server, use a code page instead of an iconv encoding. To determine the correct the value for the StringEncoding property with a SQL Server database, look up the code page that corresponds to the regional settings of the server's operating system.

For example, to use code page 932 (Japanese Shift-JIS):

StringEncoding=CP932

Teradata

For Teradata, you must override some default behavior. Teradata supports per-column character encoding, while Campaign supports only per-data source encoding. UTF-8 cannot be used with Campaign due to a bug in the Teradata ODBC driver. Teradata sets a default character encoding for each login. You can override this using a parameter in the ODBC data source configuration on Windows or in the odbc.ini on UNIX platforms as follows:

CharacterSet=UTF8

The default encoding for a Teradata table is LATIN. Teradata has very few built-in encodings, but it supports user-defined encodings.

The default value of the StringEncoding property is ASCII.

Important: For many situations involving a UTF-8 database, you should use WIDEUTF-8 pseudo-encoding, described in the WIDEUTF-8 section.

WIDEUTF-8

Campaign is normally responsible for transcoding between its internal encoding, UTF-8, and the encoding of the database. When the database is encoded in UTF-8, the value UTF-8 can be specified for StringEncoding (except for SQLServer), and no transcoding will be needed. Traditionally, these have been the only viable models for Campaign to access non-English data within a database.

In the 7.0 version of Campaign, a new database encoding called WIDEUTF-8 was introduced as a value for the StringEncoding property. By using this encoding, Campaign still uses UTF-8 to communicate with the database client, but allows the client to perform the task of transcoding between UTF-8 and the encoding of the actual database. This enhanced version of UTF-8 is needed to alter the widths of table column mappings so that they will be wide enough for transcoded text.

Note: The WIDEUTF-8 pseudo-encoding may be used only in the database configuration. It should not be used for any other purpose.

Note: Oracle does not support transcoding through the client.

SuffixOnAllOtherSQL

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The SuffixOnAllOtherSQL property specifies a string that is automatically appended to every SQL expression, generated by Campaign, which are not covered by the SuffixOnInsertSQL, SuffixOnSelectSQL, SuffixOnTempTableCreation, SuffixOnUserTableCreation, or SuffixOnUserBaseTableCreation properties.

This property applies only to SQL generated by Campaign, and does not apply to SQL in raw SQL expressions used in the Select process.

SuffixOnAllOtherSQL is used for the following expression types, when generated by Campaign:

```
TRUNCATE TABLE table
 DROP TABLE table
 DELETE FROM table [WHERE ...]
 UPDATE table SET ...
```

This property is automatically added to the SQL expression without checking its syntax. If you use this parameter, make sure that it is a legal expression. The string may be enclosed in quotation marks, but this is not required.

This property is undefined by default.

Tokens available to **SuffixOnAllOtherSQL** are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which temp tables were created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which temp tables were created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which temp tables were created.
<DBUSER>	This token is replaced with the database user name for the database where the temp tables were created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the temp table creation.
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

No default value defined.

SuffixOnCreateDateField

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The SuffixOnCreateDateField property specifies a string that Campaign automatically appends to any DATE fields in the CREATE TABLE SQL statement.

For example, you might set this property as follows:

```
SuffixOnCreateDateField = FORMAT 'YYYY-MM-DD'
```

If this property is undefined (the default), the CREATE TABLE command is unchanged.

Note: See the table in the description of the DateFormat property.

Default value

No default value defined.

SuffixOnExtractTableCreation

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Use the SuffixOnExtractTableCreation property to specify a string that is automatically appended to the SQL expression generated by Campaign when an Extract table is created.

Tokens available to **SuffixOnExtractTableCreation** are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which the Extract table was created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which the Extract table was created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which the Extract table was created.
<DBUSER>	This token is replaced with the database user name for the database where the Extract table was created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the Extract table creation.
<KEYCOLUMNS>	This token is replaced with the Extract table column name(s).
<TABLENAME>	This token is replaced with the Extract table name.

Token	Description
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

Not defined

Valid Values

Valid SQL

SuffixOnInsertSQL

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

The `SuffixOnInsertSQL` property specifies a string that is automatically appended to all `INSERT` SQL expressions generated by Campaign. This property applies only to SQL generated by Campaign, and does not apply to SQL in raw SQL expressions used in the Select process.

`SuffixOnInsertSQL` is used for the following expression type, when generated by Campaign:

```
INSERT INTO table ...
```

This property is automatically added to the SQL expression without checking its syntax. If you use this property, make sure that it is a legal expression. The string may be enclosed in quotation marks, but this is not required.

This property is undefined by default.

Tokens available to `SuffixOnInsertSQL` are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which temp tables were created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which temp tables were created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which temp tables were created.
<DBUSER>	This token is replaced with the database user name for the database where the temp tables were created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the temp table creation.
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

No default value defined.

SuffixOnSegmentTableCreation

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Specifies a string that is automatically appended to the SQL expression generated by Campaign when a Segment temp table is created.

Tokens available to **SuffixOnSegmentTableCreation** are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which the Segment temp table was created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which the Segment temp table was created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which the Segment temp table was created.
<DBUSER>	This token is replaced with the database user name for the database where the Segment temp table was created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the Segment temp table creation.
<KEYCOLUMNS>	This token is replaced with the Segment temp table column name(s).
<TABLENAME>	This token is replaced with the Segment temp table name.
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

Not defined

Valid Values

Valid SQL

SuffixOnSelectSQL

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The SuffixOnSelectSQL property specifies a string that is automatically appended to all SELECT SQL expressions generated by Campaign. This property applies only to SQL generated by Campaign, and does not apply to SQL in "raw SQL" expressions used in the Select process.

This property is automatically added to the SQL expression without checking its syntax. If you use this property, make sure that it is a legal expression. The string may be enclosed in quotation marks, but this is not required.

This property is undefined by default.

Tokens available to **SuffixOnSelectSQL** are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which temp tables were created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which temp tables were created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which temp tables were created.
<DBUSER>	This token is replaced with the database user name for the database where the temp tables were created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the temp table creation.
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

No default value defined.

SuffixOnSnapshotTableCreation

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

Use the `SuffixOnSnapshotTableCreation` property to specify a string that is automatically appended to the SQL expression generated by Campaign when a Snapshot table is created.

Tokens available to **SuffixOnSnapshotTableCreation** are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which the Snapshot table was created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which the Snapshot table was created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which the Snapshot table was created.

Token	Description
<DBUSER>	This token is replaced with the database user name for the database where the Snapshot table was created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the Snapshot table creation.
<TABLENAME>	This token is replaced with the Snapshot table name.
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

Not defined

Valid Values

Valid SQL

SuffixOnTempTableCreation

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

Use this property to specify a string that is automatically appended to the SQL expression generated by Campaign when a temp table is created. This property applies only to SQL generated by Campaign, and does not apply to SQL in "raw SQL" expressions used in the Select process. To use this property, the `AllowTempTables` property must be set to `TRUE`.

You may want to use tokens to substitute the table name and the column name(s) (`<TABLENAME>` and `<KEYCOLUMNS>`) in this SQL statement, since these are generated dynamically during the execution of the campaign.

This property is automatically added to the SQL expression without checking its syntax. If you use this property, make sure that it is a legal expression. The string may be enclosed in quotation marks, but this is not required.

This property is undefined by default.

Note: For Oracle databases, the configuration parameter is appended to the temp table creation SQL expression after the table name.

Tokens available to `SuffixOnTempTableCreation` are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which temp tables were created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which temp tables were created.

Token	Description
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which temp tables were created.
<DBUSER>	This token is replaced with the database user name for the database where the temp tables were created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the temp table creation.
<KEYCOLUMNS>	This token is replaced with the temp table column name(s).
<TABLENAME>	This token is replaced with the temp table name.
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

No default value defined.

SuffixOnUserBaseTableCreation

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

Use the `SuffixOnUserBaseTableCreation` property to specify a string that is automatically appended to the SQL expression that Campaign generates when a user creates a Base table (for example, in an Extract process). This property applies only to SQL generated by Campaign, and does not apply to SQL in "raw SQL" expressions used in the Select process.

You may want to use tokens to substitute the table name and the column name(s) (`<TABLENAME>` and `<KEYCOLUMNS>`) in this SQL statement, since these are generated dynamically during the execution of the campaign.

This property is automatically added to the SQL expression without checking its syntax. If you use this property, make sure that it is a legal expression. The string may be enclosed in quotation marks, but this is not required.

This property is undefined by default.

Tokens available to **SuffixOnUserBaseTableCreation** are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which temp tables were created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which temp tables were created.

Token	Description
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which temp tables were created.
<DBUSER>	This token is replaced with the database user name for the database where the temp tables were created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the temp table creation.
<KEYCOLUMNS>	This token is replaced with the temp table column name(s).
<TABLENAME>	This token is replaced with the temp table name.
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

No default value defined.

SuffixOnUserTableCreation

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

Use the `SuffixOnUserTableCreation` property to specify a string that is automatically appended to the SQL expression that Campaign generates when a user creates a General table (for example, in a Snapshot process). This property applies only to SQL generated by Campaign, and does not apply to SQL in "raw SQL" expressions used in the Select process.

This property is automatically added to the SQL expression without checking its syntax. If you use this property, make sure that it is a legal expression. The string may be enclosed in quotation marks, but this is not required.

This property is undefined by default.

Tokens available to **SuffixOnUserTableCreation** are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which temp tables were created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which temp tables were created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which temp tables were created.
<DBUSER>	This token is replaced with the database user name for the database where the temp tables were created.

Token	Description
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the temp table creation.
<TABLENAME>	This token is replaced with the temp table name.

Default value

No default value defined.

SystemTableSchema

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Specifies the schema used for Campaign system tables.

The default value is blank. This parameter is only relevant for the UA_SYSTEM_TABLES data source.

This property is not required for SQL server. For other data sources, set this property to the user of the database to which you are trying to connect.

You can leave this value blank unless the UA_SYSTEM_TABLES data source contains multiple schemas (for example, an Oracle database used by multiple groups). In this context, "schema" indicates the initial portion of a "qualified" table name of the form X.Y, where X is the schema and Y is the unqualified table name. For example: dbo.UA_Folder. This terminology for this syntax differs among the different database systems supported by Campaign.)

If multiple schemas exist in the system tables database, set this value to the name of the schema in which the Campaign system tables were created.

Default value

No default value defined.

TableListSQL

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Use the TableListSQL property to specify the SQL query to use to include synonyms in the list of tables available to map.

The default value is blank. This property is required if your data source is SQL Server and you want to be able to map synonyms in the returned table schema. This property is optional if you want to use a specific SQL query with other data sources in place of, or in addition to, the table schema information retrieved using the standard methods (such as an ODBC call or native connection).

Note: To ensure that Campaign works with SQL Server synonyms, you must set the UseSQLToRetrieveSchema property to TRUE in addition to setting this property as described here.

If you set this property with a valid SQL query, IBM Campaign issues the SQL query to retrieve the list of tables for mapping. If the query returns one column, it is treated as a column of names; if the query returns two columns, the first column is assumed to be a column of owner names, and the second column is considered to be a column of table names.

If the SQL query does not begin with an asterisk (*), IBM Campaign merges this list with the list of tables that are normally retrieved (such as through ODBC calls or native connections).

If the SQL query begins with an asterisk (*), the list returned by the SQL *replaces* the normal list, rather than being merged with it.

Default value

None

Valid Values

A valid SQL query

Example

If the data source is SQL Server, under normal circumstances the ODBC API call that IBM Campaign uses returns a list of tables and views, but no synonyms. To include the list of synonyms as well, set `TableListSQL` similar to the following example:

```
select B.name AS oName, A.name AS tName
from sys.synonyms A LEFT OUTER JOIN sys.schemas B
on A.schema_id = B.schema_id ORDER BY 1, 2
```

To retrieve the list of tables, views, and synonyms, avoiding the ODBC API completely, set `TableListSQL` similar to the following example:

```
*select B.name AS oName, A.name AS tName from
 (select name, schema_id from sys.synonyms UNION
select name, schema_id from sys.tables UNION select name,
schema_id from sys.views) A LEFT OUTER JOIN sys.schemas B on
A.schema_id = B.schema_id ORDER BY 1, 2
```

If the data source is Oracle, you can use a query similar to the following to retrieve the list of tables, views, and synonyms in place of the data retrieved using the native connection method that looks at the ALL_OBJECTS view:

```
*select OWNER, TABLE_NAME from (select OWNER, TABLE_NAME
from ALL_TABLES UNION select OWNER, SYNONYM_NAME AS TABLE_NAME
FROM ALL_SYNONYMS UNION select OWNER,
VIEW_NAME AS TABLE_NAME from ALL_VIEWS) A ORDER BY 1, 2
```

TempTablePostExecutionSQL

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

Use this property to specify a complete SQL statement that IBM Campaign runs immediately after the creation of a temporary table in a user data source or in the system tables database. For example, to improve performance, you can create an index on a temporary table immediately after its creation (see examples below). To enable the creation of temporary tables in a data source, the `AllowTempTables` property must be set to `TRUE`.

You can use tokens to substitute the table name (<TABLENAME>) and column names (<KEYCOLUMNS>) in the SQL statement, because the values are generated dynamically when the campaign runs.

This property is automatically added to the SQL expression without checking its syntax. If you use this property, make sure that it is a legal expression. You can enclose the string in quotation marks, but this is not required.

This property treats semicolons as delimiters to run multiple SQL statements. If your SQL statement contains semicolons and you want it to run as one statement, use a backslash as an escape character before the semicolons.

Note: If you are using stored procedures with this property, be sure that you use the correct syntax for your database.

Tokens available to **TempTablePostExecutionSQL** are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which temp tables were created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which temp tables were created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which temp tables were created.
<DBUSER>	This token is replaced with the database user name for the database where the temp tables were created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the temp table creation.
<KEYCOLUMNS>	This token is replaced with the temp table column name(s).
<TABLENAME>	This token is replaced with the temp table name.
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

No default value defined.

Examples

The following value creates an index on the temp table just after its creation, to improve the data retrieval process: `CREATE INDEX IND_<TABLENAME> ON <TABLENAME> (<KEYCOLUMNS>)`

The following example for Oracle calls a stored procedure and uses backslashes to escape the semicolon: `begin dbms_stats.collect_table_stats()\; end\;`

TempTablePrefix

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property specifies a string that is automatically prepended to the names of all temporary tables created by Campaign. Use this property to help identify and manage temp tables. You also can use this property to cause temp tables to be created in a particular location.

For example, if the user token corresponds to a schema, you can set

```
TempTablePrefix="<USER>"
```

and all temp tables will be created in the schema of whatever user is connected to the data source.

If two or more data sources point to the same database, errors and incorrect search results can occur during flowchart runs due to usage of the same temporary tables by different process boxes and flowcharts. This situation can also occur with Extract process tables and Strategic Segment tables. To avoid this situation, use `TempTablePrefix` (or `ExtractTablePrefix` for Extract tables) to define different schemas for each data source. This approach ensures that the initial part of the name is different, so the table names will always be different.

For example, give each data source a unique TempTablePrefix such as UAC_DS1 and UAC_DS2 to distinguish between temp tables for each data source. The same concept applies if you are sharing data source schemas. For example, the following prefixes allow the temp tables to be unique for both data sources that write temp tables to the same database:

DS1 TempTablePreFix: schemaA.UAC_DS1

DS2 TempTablePreFix: schemaA.UAC_DS2

The following table describes the tokens that are available to **TempTablePrefix**.

Note: You must make sure that the final temp table name after resolving tokens does not exceed any database-specific name length restrictions.

Note: In tokens used for TempTablePrefix, any characters that are not valid for database table names will be stripped. After tokens are resolved, the resulting temp table prefixes must start with an alphabetic character, and must contain only alphanumeric characters or underscore characters. Illegal characters will be removed silently. If any resulting temp table prefix does not begin with an alphabetic character, Campaign prepends the letter "U" to the prefix.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which temp tables were created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which temp tables were created.

Token	Description
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which temp tables were created.
<DBUSER>	This token is replaced with the database user name for the database where the temp tables were created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the temp table creation.
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

UAC

TempTablePreTruncateExecutionSQL

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Note: This property is supported only for Teradata data sources. For all other supported databases, this property should not be set.

Use this property to specify a SQL query to run before a temp table is truncated. The query that you specify can be used to negate the effect of a SQL statement specified in the **TempTablePostExecutionSQL** property.

For example, with the **TempTablePostExecutionSQL** property, you could specify the following SQL statement to create an index:

```
CREATE INDEX <TABLENAME>Idx_1 (<KEYCOLUMNS>) ON <TABLENAME>
```

Then, specify the following query in the **TempTablePreTruncateExecutionSQL** property to drop the index:

```
DROP INDEX <TABLENAME>Idx_1 ON <TABLENAME>
```

Default value

Not defined

Valid Values

A valid SQL query

TempTablePreTruncateRunScript

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Note: This property is supported only for Teradata data sources. For all other supported databases, this property should not be set.

Use this property to specify a script or executable to run before a temp table is truncated. The script that you specify can be used to negate the effect of a SQL statement specified in the **PostTempTableCreateRunScript** property.

For example, with the **PostTempTableCreateRunScript** property, you could specify a script that includes the following SQL statement to create an index:

```
CREATE INDEX <TABLENAME>Idx_1 (<KEYCOLUMNS>) ON <TABLENAME>
```

Then, specify another script with the following statement in the **TempTablePreTruncateRunScript** property to drop the index:

```
DROP INDEX <TABLENAME>Idx_1 ON <TABLENAME>
```

Default value

Not defined

Valid Values

File name of a shell script or executable

TeradataDeleteBeforeDrop

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

This property applies only to Teradata data sources. It specifies whether records are deleted before a table is dropped.

Set the value to `TRUE` to delete all records from a table before dropping the table.

Note: If IBM Campaign is unable to delete the records for any reason, it will not drop the table.

Set the value to `FALSE` to drop a table without first deleting all records.

Default value

`TRUE`

TruncateSQL

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

This property is available for use with DB2 data sources, and allows you to specify alternate SQL for table truncation. This property applies only when **DeleteAsTruncate** is TRUE. When **DeleteAsTruncate** is TRUE, any custom SQL in this property is used to truncate a table. When this property is not set, IBM Campaign uses the TRUNCATE TABLE <TABLENAME> syntax.

This property is undefined by default.

Tokens available to **TruncateSQL** are described below.

Token	Description
<TABLENAME>	This token is replaced with the database table name that IBM Campaign is truncating.

Default value

No default value defined.

Type

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property specifies the database type of this data source.

Default value

The default value depends on the database template that was used to create the data source configuration.

Valid Values

Valid Types for system tables are:

- DB2
- DB2ODBC
- ORACLE
- ORACLE8
- ORACLE9
- SQLServer

Valid Types for customer tables are:

- BigDataODBC_Hive
- DB2
- DB2ODBC
- NETEZZA
- ORACLE
- ORACLE8
- ORACLE9
- PostgreSQL
- SQLServer
- TERADATA

UOSQLOnConnect

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The SQLOnConnect property defines a complete SQL statement that Campaign runs immediately after each database connection. The UOSQLOnConnect property is similar to this, but specifically applicable to Contact Optimization.

The SQL statement generated by this property is automatically passed to your database without checking its syntax. If you use this property, make sure that it is a legal expression. The string may be enclosed in quotation marks, but this is not required.

This property is undefined by default.

Tokens available to **UOSQLOnConnect** are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which temp tables were created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which temp tables were created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which temp tables were created.
<DBUSER>	This token is replaced with the database user name for the database where the temp tables were created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the temp table creation.
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

No default value defined.

UseAliasForPredicate

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

This property is used only for Hive-based Hadoop data sources (BigDataODBCHiveTemplate). Set the value to TRUE if you are connecting to IBM BigInsight Hadoop instance. Set to FALSE if you are connecting to any other Hive based Hadoop instance.

Note: If you are upgrading to version 11.0 and you have already configured and are using a Hive based Hadoop data source, you do not need to make any changes for the existing instance.

Default value

`FALSE`

Valid Values

`TRUE | FALSE`

UseExceptForMerge

Configuration category

`Campaign|partitions|partition[n]|dataSources|`*`dataSourcename`*

Description

When IBM Campaign performs exclusions in the Merge process or in the Segment process, by default it uses NOT EXISTS syntax, as:

```
SELECT IncludeTable.ID FROM IncludeTable WHERE NOT EXISTS
(SELECT * FROM ExcludeTable WHERE IncludeTable.ID = ExcludeTable.ID)
```

If **UseExceptForMerge** is TRUE and you cannot use NOT IN (because **UseNotInForMerge** is disabled or because the audience level consists of multiple fields and the data source is not Oracle), then the syntax is altered as follows:

Oracle

```
SELECT IncludeTable.ID FROM IncludeTable
MINUS (SELECT ExcludeTable.ID FROM ExcludeTable)
```

Others

```
SELECT IncludeTable.ID FROM IncludeTable
EXCEPT (SELECT ExcludeTable.ID FROM ExcludeTable)
```

For Hive-based Hadoop big data, this property must be FALSE. Hive does not support the EXCEPT clause, so a setting of TRUE can result in process failures.

Default value

FALSE

Valid Values

TRUE | FALSE

UseGroupByForDistinct

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The **UseGroupByForDistinct** property is available for the Teradata datasource template. By default, this property is FALSE. When this property is enabled, <select query> uses a GROUP BY statement instead of DISTINCT.

The purpose of this property: Flowcharts execute the query "select DISTINCT <audience id> from <table>" to fetch unique records when a table is not mapped as a normalized table in IBM Campaign. When such queries are submitted against Teradata, it causes additional sorting of data on the database and high CPU consumption. The Teradata DBA recommendation is to use GROUP BY instead of DISTINCT, because GROUP BY can take advantage of Teradata multi AMP processing architecture.

Default value

FALSE

Valid Values

TRUE | FALSE

UseMergeForTrack

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property implements SQL MERGE syntax to improve the performance of the Track process in flowcharts. This property can be set to TRUE for DB2, Oracle, SQL Server 2008, and Teradata 12. It can also be used with other databases that support the SQL MERGE statement.

Default value

TRUE (DB2 and Oracle) | FALSE (all others)

Valid Values

TRUE | FALSE

UseNonANSIJoin

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

The UseNonANSIJoin property specifies whether this data source uses non-ANSI join syntax. If the data source type is set to Oracle7 or Oracle8, and the value of UseNonANSIJoin is set to TRUE, the data source uses non-ANSI join syntax appropriate for Oracle.

Default value

FALSE

Valid Values

TRUE | FALSE

UseNotInForMerge

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

When IBM Campaign performs exclusions in the Merge process or in the Segment process, by default it uses NOT EXISTS syntax, as:

```
SELECT IncludeTable.ID FROM IncludeTable WHERE NOT EXISTS (SELECT *
FROM ExcludeTable WHERE IncludeTable.ID = ExcludeTable.ID)
```

If **UseNotInForMerge** is enabled and either (1) the audience level is composed of a single ID field, or (2) the data source is Oracle, then the syntax is altered as follows:

```
SELECT IncludeTable.ID FROM IncludeTable WHERE IncludeTable.ID NOT IN
(SELECT ExcludeTable.ID FROM ExcludeTable)
```

Default value

FALSE

Valid Values

TRUE | FALSE

UseNotInToDeleteCH

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property affects the IBM Campaign system table data source (UA_SYSTEM_TABLES). It affects the SQL query syntax for how the MailList and CallList processes remove records from the IBM Campaign system tables.

The default value of FALSE typically improves database performance. The default behavior uses EXISTS / NOT EXISTS when removing Contact History records (either after a failed run or in response to the user's action in the GUI). The removal process involves deleting from UA_OfferHistAttrib and updating UA_OfferHistory.

You can change this value to TRUE if you prefer to use the SQL syntax of IN / NOT IN. Prior versions of IBM Campaign used IN / NOT IN.

Default value

FALSE

Valid Values

TRUE | FALSE

UserBaseTablePostExecutionSQL

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property is invoked when a process box is configured to write to a **New Mapped Table** > **Base Record Table** > **Create New Table in Selected Database**. This property is invoked only when the table is created (during the creation and mapping process). This property is not invoked during process box runtime.

This property is automatically added to the SQL expression without checking its syntax. If you use this property, make sure that it is a legal expression. You can enclose the string in quotation marks, but this is not required.

This property treats semicolons as delimiters to run multiple SQL statements. If your SQL statement contains semicolons and you want it to run as one statement, use a backslash as an escape character before the semicolons.

Note: If you are using stored procedures with this property, be sure to use the correct syntax for your database. The following example for Oracle calls a stored procedure and uses backslashes to escape the semicolon:
`begin dbms_stats.collect_table_stats()\; end\;`

You can use tokens to substitute the <TABLENAME> in this SQL statement because the name is generated dynamically when the campaign runs. For available tokens, see **UserTablePostExecutionSQL**.

UserTablePostExecutionSQL

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

Use this property to specify a complete SQL statement that IBM Campaign runs immediately after the creation of a user table in a user data source or in the system tables database. This property is invoked when a process box writes to one of the following tables:

- **New mapped table** > **General table** > **Create new table in selected datasource**: The property is invoked during the creation/mapping process; not during Snapshot runtime.
- **New mapped rable** > **Dimension table** > **Create new table in selected database**: The property is invoked during the creation/mapping process; not during Snapshot runtime.
- **Database table**: The property is invoked during process box runtime.

This property is automatically added to the SQL expression without checking its syntax. If you use this property, make sure that it is a legal expression. You can enclose the string in quotation marks, but this is not required.

This property treats semicolons as delimiters to run multiple SQL statements. If your SQL statement contains semicolons and you want it to run as one statement, use a backslash as an escape character before the semicolons.

Note: If you are using stored procedures with this property, be sure to use the correct syntax for your database. The following example for Oracle calls a stored procedure and uses backslashes to escape the semicolon:
`begin dbms_stats.collect_table_stats()\; end\;`

You can use tokens to substitute the `<TABLENAME>` in this SQL statement, because the name is generated dynamically when the campaign runs.

Tokens available to **UserTablePostExecutionSQL** are described below.

Token	Description
<AMUSER>	This token is replaced with the IBM Marketing Software user name associated with the flowchart for which the user tables were created.
<CAMPAIGNCODE>	This token is replaced with the code for the campaign associated with the flowchart for which the user tables were created.
<CAMPAIGNNAME>	This token is replaced with the name of the campaign associated with the flowchart for which the user tables were created.
<DBUSER>	This token is replaced with the database user name for the database where the user tables were created.
<FLOWCHARTNAME>	This token is replaced with the name of the flowchart associated with the user table creation.
<TABLENAME>	This token is replaced with the user table name.

Token	Description
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

Default value

No default value defined.

UseSQLToProfile

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property allows you to configure IBM Campaign to submit the SQL query GROUP BY to the database to compute profiles (using SELECT *field*, count(*) FROM *table* GROUP BY *field*), rather than fetching records.

- A value of FALSE (the default) causes IBM Campaign to profile a field by retrieving the field value for all records in the table and to track the count of each distinct value.
- A value of TRUE causes IBM Campaign to profile a field by issuing a query similar to the following:

```
SELECT field, COUNT(*) FROM table GROUP BY field
```

which pushes the burden to the database.

Default value

FALSE

Valid Values

TRUE | FALSE

UseSQLToRetrieveSchema

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

This property determines whether Campaign uses a SQL query, rather than an ODBC or native API call, to retrieve the schema to use as the table schema for this data source.

The default value is FALSE, indicating that Campaign should use its standard method (ODBC or native connection, for example) to retrieve the schema. Setting this property to TRUE causes Campaign to prepare a SQL query similar to select * from <table> to retrieve the table schema.

This can provide advantages that are specific to each data source. For example, some data sources (Netezza, SQL Server) do not properly report SQL synonyms (alternative names for database objects, defined using the create synonym syntax) through the default ODBC or native connections. By setting this property to TRUE, SQL synonyms are retrieved for data mapping within Campaign.

The following list describes the behavior of this setting for a number of data sources:

- Hive-based Hadoop big data: Use the default setting of FALSE.

- Netezza: Set this property to TRUE to allow support for SQL synonyms. No other settings or values are needed to support synonyms in Netezza data sources.
- SQL Server: To allow support for synonyms, set this property to TRUE **and** enter valid SQL in the TableListSQL property for this data source. See the description for the TableListSQL property for more details.
- Oracle: Set this property to TRUE to tell Campaign to prepare the SQL query to retrieve the table schema. The result set identifies NUMBER fields (no precision/scale specified, which may cause issues in Campaign) as NUMBER(38), which avoids those possible issues.
- For other data sources, you can optionally set this property to TRUE to use the default SQL select query described above, or to specify valid SQL in the TableListSQL property to use instead of, or in addition to, the ODBC API or native connection that is used by default. See the description for the TableListSQL property for more details.

Default value

FALSE

Valid Values

TRUE | FALSE

Example

To allow Campaign to work with Netezza or SQL Server synonyms:

UseSQLToRetrieveSchema=TRUE

UseTempTablePool

Configuration category

Campaign|partitions|partition[n]|dataSources|*dataSourcename*

Description

When UseTempTablePool is set to FALSE, temp tables are dropped and re-created every time a flowchart is run. When the property is set to TRUE, temp tables are not dropped from the database. Temp tables are truncated and reused from the pool of tables maintained by Campaign. The temp table pool is most effective in environments where you rerun flowcharts many times, such as during a design and test phase.

Default value

FALSE

Valid Values

TRUE | FALSE

Campaign | partitions | partition[n] | systemTableMapping

Properties in the systemTableMapping category are populated automatically if you remap any system tables or map Contact or Response history tables. You should not edit properties in this category.

Campaign | partitions | partition[n] | server

This category contains properties to configure the IBM Campaign server for the selected partition.

Campaign | partitions | partition[n] | server | systemCodes

Properties in this category specify, for Campaign, whether variable length codes are allowed, the format and generator of the campaign and cell codes, whether offer codes are displayed, and the offer code delimiter.

offerCodeDelimiter

Configuration category

Campaign|partitions|partition[n]|server|systemCodes

Description

The offerCodeDelimiter property is used internally to concatenate multiple code parts (for example, to output the OfferCode field in Campaign Generated Fields) and for incoming offer codes in the Campaign Response process, to split the offer code into multiple parts. The value must be only a single character.

Older versions of Campaign included a NumberOfOfferCodesToUse parameter. However, in more recent versions, this value comes from the offer template (every offer template can have a different number of offer codes).

Default value

-

allowVariableLengthCodes

Configuration category

Campaign|partitions|partition[n]|server|systemCodes

Description

The allowVariableLengthCodes property specifies whether variable length codes are allowed in Campaign.

If the value is TRUE, and if the trailing part of the code format is x, the length of the code can vary. For example, if the code format is nnnnxxxx, then the code can be from 4 to 8 characters long. This applies to campaign, offer, version, tracking, and cell codes.

If the value is FALSE, variable length codes are not allowed.

Default value

FALSE

Valid Values

TRUE | FALSE

displayOfferCodes

Configuration category

Campaign|partitions|partition[n]|server|systemCodes

Description

The displayOfferCodes property specifies whether to show offer codes beside their names in the Campaign GUI.

If the value is TRUE, offer codes are displayed.

If the value is FALSE, offer codes are not displayed.

Default value

FALSE

Valid Values

TRUE | FALSE

cellCodeFormat

Configuration category

Campaign|partitions|partition[n]|server|systemCodes

Description

The cellCodeFormat property is used by the campaign code generator to define the format of the cell code that is automatically created by the default cell code generator. For a list of valid values, see campCodeFormat.

Default value

Annnnnnnnn

campCodeFormat

Configuration category

Campaign|partitions|partition[n]|server|systemCodes

Description

The campCodeFormat property is used by the campaign code generator to define the format of the campaign code that is automatically generated by the default campaign code generator when you create a campaign.

Default value

Cnnnnnnnnn

Valid Values

The possible values are:

- A-Z or any symbol - treated as a constant
- a - random letters A-Z (upper case only)
- c - random letters A-Z or numbers 0-9
- n - random digit 0-9
- x - any single ASCII character from 0-9 or A-Z. You can edit the generated campaign code and replace the ASCII character that Campaign substituted for the x with any ASCII character, and Campaign will use that character instead.

cellCodeGenProgFile

Configuration category

Campaign|partitions|partition[n]|server|systemCodes

Description

The cellCodeGenProgFile property specifies the name of the cell code generator. The properties that control the format of the code generated are set in the cellCodeFormat property. See campCodeGenProgFile for a list of supported options.

If you write your own cell code generator, replace the default value with the absolute path of your custom program, including the file name and extension, and using forward slashes (/) for UNIX and backslashes (\) for Windows.

Default value

uaccampcodegen (the code generator supplied by Campaign)

campCodeGenProgFile

Configuration category

Campaign|partitions|partition[n]|server|systemCodes

Description

This property specifies the name of the campaign code generator. The properties that control the format of the generated code are set in the campCodeFormat property.

If you write your own campaign code generator, replace the default value with the absolute path of your custom program, including the file name and extension, using forward slashes (/) for UNIX and backslashes (\) for Windows.

The default campaign code generator can be called with the following options:

- -y Year (four integers)
- -m Month (one or two integers, cannot exceed value of twelve)
- -d Day (one or two integers, cannot exceed value of 31)
- -n Campaign name (any string, cannot exceed 64 characters)
- -o Campaign owner (any string, cannot exceed 64 characters)
- -u Campaign code (any integer). Allows you to specify the exact campaign ID rather than having the application generate one for you.
- -f Code format if overriding the default. Takes the values specified in campCodeFormat.
- -i Other integer.
- -s Other string.

Default value

uaccampcodegen (the code generator supplied by Campaign)

cellCodeBulkCreation

Configuration category

Campaign|partitions|partition[n]|server|systemCodes

Description

A value of TRUE improves performance of the cell code generation utility during bulk creation of cell codes, because multiple cell codes are generated with a single invocation of the cell code generator. This is more efficient and is the recommended setting. A value of TRUE also improves performance when copying flowcharts, templates, and process boxes.

When the value is FALSE, the cell code generator is invoked once for each cell code generation. If cell code generation seems to take a long time for Segment, Sample, and Decision process boxes, or for the target cell spreadsheet, set this value to TRUE.

The default setting is FALSE to support existing customized implementations. If you are using a legacy custom-made cell code generation utility, leave this setting at its default value of FALSE until you implement a new custom utility. Then you can change its value to TRUE.

If you are not using a custom cell code generation utility, change the value to TRUE to take advantage of the efficiency improvements.

Default value

FALSE

Valid Values

TRUE | FALSE

Campaign | partitions | partition[n] | server | encoding

The property in this category specifies the text encoding for values written to files, to support non-English data.

stringEncoding

Description

The partition[n] > server> encoding > stringEncoding property how Campaign reads in and writes out flat files. It should match the encoding used for all flat files. If not configured elsewhere, this is the default setting for flat file encoding.

Note: WIDEUTF-8 is not supported for this setting.

By default, no value is specified, and outgoing text files are encoded as UTF-8, which is the default encoding for Campaign.

It is a best practice to explicitly set this value to an encoding appropriate for your system, even if the value is UTF-8, the same as the implicit default.

Note: If you do not set the value of the StringEncoding property for data sources in the dataSources category, the value of this stringEncoding property is used as the default value. This can cause unnecessary confusion -- you should always explicitly set the StringEncoding property in the dataSources category.

See the *Campaign Administrator's Guide* for a list of supported encodings.

Default value

No default value defined.

forceDCTOneBytePerChar

Description

The forceDCTOneBytePerChar property specifies whether Campaign should use the original field width for output files, rather than the potentially expanded width reserved to allow sufficient space for transcoding into UTF-8.

A text value may have different lengths, depending on the encoding used to represent it. When the text value comes from a data source whose stringEncoding property is neither ASCII nor UTF-8, Campaign reserves three times the field width in order to ensure sufficient space for

transcoding into UTF-8. For example, if the stringEncoding property is set to LATIN1, and the field in the database is defined as VARCHAR(25), Campaign will reserve 75 bytes to hold the transcoded UTF-8 value. Set the forceDCTOneBytePerChar property to TRUE if you want to use the original field width.

Default value

FALSE

Valid Values

TRUE | FALSE

Campaign | partitions | partition[n] | server | timeout

The properties in this category specify the number of seconds an Campaign flowchart waits, after the user has disconnected and all runs have completed, before exiting, and the Campaign server process waits for a response from external servers before reporting an error.

waitForGracefulDisconnect

Description

The waitForGracefulDisconnect property specifies whether the Campaign server process continues to run until the user gracefully disconnects, or exits regardless of whether the user intended to disconnect.

If the value is yes, the default, the server process continues to run until it can determine that the user wants it to exit. This option prevents changes from being lost, but can result in server processes accumulating.

If the value is no, the server process shuts down and server processes are prevented from accumulating, but users can lose work if a network interruption occurs or if they do not follow the recommended sequence of actions to exit gracefully.

Default value

TRUE

Valid Values

TRUE| FALSE

urlRequestTimeout

Description

The urlRequestTimeout property specifies the number of seconds the Campaign server process waits for a response from external servers. Currently, this applies to requests to IBM Marketing Software servers and eMessage components that operate with Campaign.

If the Campaign server process does not receive a response within this period, a communication timeout error is reported.

Default value

60

delayExitTimeout

Description

The delayExitTimeout property specifies the number of seconds an Campaign flowchart waits, after the user has disconnected and all runs have completed, before exiting.

Setting this property to a non-0 value enables subsequent Campaign flowcharts to make use of existing instances rather than starting a new instance.

Default value

10

Campaign | partitions | partition[n] | server | collaborate

This category applies to IBM Distributed Marketing.

collaborateInactivityTimeout

Configuration category

Campaign|partitions|partition[n]|server|collaborate

Description

The collaborateInactivityTimeout property specifies the number of seconds the unica_acsvr process waits after it finishes servicing a Distributed Marketing request before it exits. This waiting period allows the process to remain available in the typical scenario in which Distributed Marketing makes a series of requests prior to running the flowchart.

The minimum value is 1. Setting this property to 0 causes it to default to 60.

Default value

60

logToSeparateFiles

Configuration category

Campaign|partitions|partition[n]|server|collaborate

Description

This property was introduced in v8.6.0.6. By default and upon upgrade, the value of this parameter is False.

When True, flowchart logs for runs initiated from Distributed Marketing are logged to separate log files. Log files are created under folders with the current date to avoid an excessive number of log files in a single folder. The format of the folder name is "FlowchartRunLogs_<YYYYMMDD>".

The format of the log file name is: <CAMP_NAME>_<CAMP_CODE>_<FC_NAME>_<PID>_<LIST_CODE> _<DATE>_<TIMESTAMP>.log, where PID is the Campaign server process ID that ran the flowchart. LIST_CODE is the object code of the Distributed Marketing list, ONDC or corporate campaign from which the flowchart was run.

All user variables that are passed to the flowchart run process are logged for troubleshooting purposes.

Note: When a flowchart is opened, it initially logs to the traditional flowchart log file. When a flowchart run is initiated from Distributed Marketing, if logToSeparateFiles is True, logging is done in the new directory and file at that time.

Default value

`False`

Valid Values

`True | False`

Campaign | partitions | partition[n] | server | spss

Properties in this category affect IBM SPSS Modeler Advantage Enterprise Marketing Management Edition integration for the specified partition in IBM Campaign.

SharedDirectoryPathCampaign

Description

The path to the directory that is used to transfer data between IBM Campaign and IBM SPSS Modeler Server, as seen from IBM Campaign.

- IBM Campaign puts input data files to IBM SPSS Modeler Advantage Enterprise Marketing Management Edition in this directory.
- IBM SPSS Modeler Advantage Enterprise Marketing Management Edition puts output data files in this directory to be read and processed by IBM Campaign.

Default value

None

Valid values

Any valid Windows path (such as `Z:\SPSS_Shared`) or a mount directory (for UNIX).

SharedDirectoryPathSPSS

Description

The path to the directory that is used to transfer data between IBM Campaign and IBM SPSS Modeler Server, as seen from IBM SPSS Modeler Server. This is the same shared directory referenced by `SharedDirectoryPathCampaign` but is the local directory path used by IBM SPSS Modeler Server.

For example, IBM Campaign might be installed on Windows with `SharedDirectoryPathCampaign = Z:\SPSS_Shared`, where `Z:\SPSS_Shared` is a mapped network drive, while IBM SPSS Modeler Server is installed on UNIX with a mount to that directory defined as `SharedDirectoryPathSPSS = /share/CampaignFiles`.

Default value

None

Valid values

Any valid Windows path (such as `Z:\SPSS_Shared`) or a mount directory (such as `/share/CampaignFiles`) for UNIX.

C&DS_URL

Description

The URL for the IBM SPSS Collaboration and Deployment Services repository.

Default value

`http://localhost:7001/cr-ws/services/ContentRepository`

Valid values

The URL for the IBM SPSS Collaboration and Deployment Services repository.

SPSS_Integration_Type

Description

This property determines the type of integration between IBM Campaign and IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

Default value

`None`

Valid values

- `None`: No integration
- `SPSS MA Marketing Edition`: Full integration of modeling and scoring. This option is only available if IBM SPSS Modeler Advantage Enterprise Marketing Management Edition is installed and configured.
- `Scoring only`: Scoring, but not modeling, is enabled.

Campaign | partitions | partition[n] | server | permissions

The properties in this category specify the permissions for folders that are created by Campaign and the UNIX group and permissions for files in the `profile` directory.

userFileGroup (UNIX only)

Description

This property specifies a group associated with user-generated Campaign files. The group will be set only if the user is a member of the specified group.

This property is undefined by default.

Default value

No default value defined.

createFolderPermissions

Description

The `createFolderPermissions` parameter specifies the permissions of directories that are created by Campaign on the Campaign server (partition[n] location) by using the Create Folder icon on the Open Data Source File dialog in the table mapping.

Default value

755 (owner has read/write/execute access, group and world have execute/read access)

catalogFolderPermissions

Description

The catalogFolderPermissions property specifies the permissions of directories created by Campaign through the Stored table catalogs > Create folder window.

Default value

755 (owner has read/write/execute access, group and world have execute/read access)

templateFolderPermissions

Description

The templateFolderPermissions property specifies the permissions of template directories created by Campaign through the **Stored templates > Create folder** window.

Default value

755 (owner has read/write/execute access, group and world have read/execute access)

adminFilePermissions (UNIX only)

Description

The adminFilePermissions property specifies a permission bit mask for the files contained in the profile directory.

Default value

660 (owner and group have read/write access only)

userFilePermissions (UNIX only)

Description

The userFilePermissions property specifies a permission bit mask for user generated Campaign files (for example, log files, summary files, exported flat files).

Default value

666 (everyone can read and write files created by Campaign in the server)

adminFileGroup (UNIX only)

Description

The adminFileGroup property specifies a UNIX admin group associated with files contained in the profile directory.

This property is undefined by default.

Default value

No default value defined.

Campaign | partitions | partition[n] | server | flowchartConfig

Properties in this category specify the behavior of the Campaign Generated Field, whether duplicate cell codes are allowed, and whether the Log to Contact History option defaults to enabled.

allowDuplicateCellcodes

Description

The `allowDuplicateCellcodes` property specifies whether the cell codes in the Campaign Snapshot process can have duplicate values.

If the value is `FALSE`, the Campaign server enforces unique cell codes.

If the value is `TRUE`, the Campaign server does not enforce unique cell codes.

Default value

`TRUE`

Valid Values

`TRUE | FALSE`

allowResponseNDaysAfterExpiration

Description

The `allowResponseNDaysAfterExpiration` property specifies the maximum number of days after all offer expiration dates that responses can tracked. These late responses can be included in performance reports.

Default value

`90`

agfProcessnameOutput

Description

The `agfProcessnameOutput` property specifies the output behavior of the Campaign Generated Field (UCGF) in the List, Optimize, Response, and Snapshot processes.

If the value is `PREVIOUS`, the UCGF contains the process name associated with the incoming cell.

If the value is `CURRENT`, the UCGF holds the process name of the process in which it is used.

Default value

`PREVIOUS`

Valid Values

`PREVIOUS | CURRENT`

logToHistoryDefault

Description

The `logToHistoryDefault` property specifies whether the Log to Contact History and Tracking Tables option in the Log tab of the Campaign contact processes defaults to enabled.

If the value is TRUE, the option is enabled.

If the value is FALSE, the option is disabled in any newly created contact processes.

Default value

TRUE

Valid Values

TRUE | FALSE

overrideLogToHistory

Description

This property controls whether users with the appropriate permissions can change the Log to Contact History Tables setting when they configure a contact or Track process. To make all flowchart production runs always write to contact history, enable logToHistoryDefault and disable overrideLogToHistory

Default value

TRUE

Valid Values

TRUE | FALSE

defaultBehaviorWhenOutputToFile

Description

Specifies the behavior for contact processes in Campaign when outputting to a file. This property applies only within the current partition. This default behavior (if set) is only applied for processes when they are newly added to flowcharts; once a process is added to a flowchart, the output behavior can be changed in the process configuration.

Default value

Replace All Records

Valid Values

- Append to Existing Data
- Create New File
- Replace All Records

defaultBehaviorWhenOutputToDB

Description

Specifies the behavior for contact processes in Campaign when outputting to a database table. This property applies only within the current partition. This default behavior (if set) is only applied for processes when they are newly added to flowcharts; once a process is added to a flowchart, the output behavior can be changed in the process configuration.

Default value

Replace All Records

Valid Values

- Append to Existing Data

- Replace All Records

replaceEmbeddedNames

Description

When replaceEmbeddedNames is TRUE, Campaign replaces user variable and UCGF names embedded in query text with actual values, although these names must be separated by a non-alphanumeric character, such as an underscore (for example, ABC_UserVar.v1 will be substituted but ABCUserVar.v1 will not). Set this property to TRUE for backwards compatibility with Campaign 7.2 and earlier.

When set to FALSE, Campaign replaces only distinct user variable and UCGF names with actual values (in both IBM Marketing Software and raw SQL expressions). Set this property to FALSE for backwards compatibility with Campaign 7.3 and higher.

Default value

FALSE

Valid Values

TRUE | FALSE

legacyMultifieldAudience

Description

In most cases, you can leave this property set to the default value of FALSE. Campaign v8.5.0.4 and newer name multifield Audience ID fields according to the audience definition, regardless of the source of the fields. When you configure processes to use multifield Audience ID fields, you now see the new Audience ID naming convention for multifield audiences. Already-configured processes in flowcharts created in previous Campaign versions should continue to work. However, if old flowcharts fail because of the change in the naming convention, you can revert Campaign behavior by setting this property toTRUE.

Default value

FALSE

Valid Values

TRUE | FALSE

Campaign | partitions | partition[n] | server | flowchartSave

The properties in this category specify the default settings for the auto-save and checkpoint properties of a new Campaign flowchart.

checkpointFrequency

Description

The checkpointFrequency property specifies (in minutes) the default setting for a new Campaign flowchart's checkpoint property, configurable for each flowchart through the client-side Advanced Settings window. The checkpoint feature provides the ability to capture a snapshot of a running flowchart for recovery purposes.

Default value

0 (zero)

Valid Values

Any integer

autosaveFrequency

Description

The autosaveFrequency property specifies (in minutes) the default setting for a new Campaign flowchart's auto-save property, configurable for each flowchart through the client-side Advanced Settings window. The auto-save function performs a forced save of flowcharts during editing and configuration.

Default value

0 (zero)

Valid Values

Any integer

Campaign | partitions | partition[n] | server | dataProcessing

Properties in this category specify how IBM Campaign handles string comparisons and empty fields in flat files, and the behavior of the macro STRING_CONCAT.

longNumericIdsAsText

Description

The longNumericIdsAsText property specifies whether the Campaign macro language will treat numeric IDs longer than 15 digits as text. This property affects ID fields. It has no effect on non-ID fields. This property is useful if you have numeric ID fields with more than 15 digits AND you want to include ID values in criteria.

- Set the value to TRUE to specify that numeric IDs longer than 15 digits will be treated as text.
- When the value is FALSE, numeric IDs longer than 15 digits are treated as numeric values (and thus might lose precision or uniqueness if truncated or rounded). If you do anything that treats the ID values as numeric (such as profiling, or using in a Derived Field), the text is converted to numeric, and precision beyond the 15th digit is lost.

Note: For non-ID numeric fields, if you do anything that treats the value as numeric (such as profiling, rounding, or using in a Derived Field), precision beyond 15 digits is lost.

This setting is ignored if the partitions > partition[n] > dataSources > [data_source_name] > ForceNumeric property is set to TRUE for fields coming from this data source.

Default value

FALSE

Valid Values

TRUE | FALSE

stringConcatWithNullIsNull

Description

The stringConcatWithNullIsNull property controls the behavior of the Campaign macro STRING_CONCAT.

When the value is TRUE, STRING_CONCAT returns NULL if any of its inputs is NULL.

When the value is FALSE, STRING_CONCAT returns the concatenation of all of its non-NULL properties; in this case, STRING_CONCAT returns NULL only if all of its inputs are NULL.

Default value

TRUE

Valid Values

TRUE | FALSE

performCaseInsensitiveComparisonAs

Description

The performCaseInsensitiveComparisonAs property specifies how Campaign compares data values when the compareCaseSensitive property is set to no (that is, during case-insensitive comparisons). This property is ignored if the value of compareCaseSensitive is yes.

When the value is UPPER, Campaign converts all data to upper case before comparing.

When the value is LOWER, Campaign converts all data to lower case before comparing.

Default value

LOWER

Valid Values

UPPER | LOWER

upperAllowsDate

Description

The upperAllowsDate property specifies whether the UPPER database function allows a DATE/DATETIME parameter, and therefore whether the operation may be performed in the database or must be performed by the Campaign server.

Set the value to TRUE if the database is SQL Server or Oracle. The UPPER function in these databases allows a DATE/DATETIME parameter.

Set the value to FALSE if the database is DB2 or Teradata. The UPPER function in these databases does not allow a DATE/DATETIME parameter.

Note that this setting is global, not per data source. If a value of no is recommended for any data source in use, set the value to no. If a value of yes is recommended for all data sources in use, set the value to yes.

Default value

TRUE

Valid Values

TRUE | FALSE

compareCaseSensitive

Description

The compareCaseSensitive property specifies whether the Campaign data comparisons are sensitive to alphabetic case (UPPER vs. lower).

When the value is FALSE, Campaign ignores case differences when comparing data values and sorts textual data in a binary, case-insensitive manner. This setting is strongly recommended when English data is used.

When the value is TRUE, Campaign distinguishes data values based on case differences, performing a true binary-value comparison of each character. This setting is strongly recommended when non-English data is used.

Default value

FALSE

Valid Values

TRUE | FALSE

lowerAllowsDate

Description

The lowerAllowsDate property specifies whether the LOWER database function allows a DATE/DATETIME parameter, and therefore whether the operation may be performed in the database or must be performed by the Campaign server.

Set the value to TRUE if the database is SQL Server or Oracle. The LOWER function in these databases allows a DATE/DATETIME parameter.

Set the value to FALSE if the database is DB2 or Teradata. The LOWER function in these databases does not allow a DATE/DATETIME parameter.

Note that this setting is global, not per data source. If a value of no is recommended for any data source in use, set the value to no. If a value of yes is recommended for all data sources in use, set the value to yes. Typically, only one database type is in use at a customer site, but there are some installations in which multiple database types are in use.

Default value

TRUE

Valid Values

TRUE | FALSE

substrAllowsDate

Description

The substrAllowsDate property specifies whether the SUBSTR/SUBSTRING database function allows a DATE/DATETIME parameter, and therefore whether the operation may be performed in the database or must be performed by the Campaign server.

Set the value to TRUE if the database is Oracle or Teradata. The SUBSTR/SUBSTRING function in these databases allows a DATE/DATETIME parameter.

Set the value to FALSE if the database is SQL Server or DB2. The SUBSTR/SUBSTRING function in these databases does not allow a DATE/DATETIME parameter.

Note that this setting is global, not per data source. If a value of no is recommended for any data source in use, set the value to no. If a value of yes is recommended for all data sources in use, set the value to yes.

Default value

TRUE

Valid Values

TRUE | FALSE

ltrimAllowsDate

Description

The ltrimAllowsDate property specifies whether the LTRIM database function allows a DATE/DATETIME parameter, and therefore whether the operation may be performed in the database or must be performed by the Campaign server.

Set the value to TRUE if the database is SQL Server, Oracle, or Teradata. The LTRIM function in these databases allows a DATE/DATETIME parameter.

Set the value to FALSE if the database is DB2. The LTRIM function in this database does not allow a DATE/DATETIME parameter.

Note that this setting is global, not per data source. If a value of no is recommended for any data source in use, set the value to no. If a value of yes is recommended for all data sources in use, set the value to yes. Typically, only one database type is in use at a customer site, but there are some installations in which multiple database types are in use.

Default value

TRUE

Valid Values

TRUE | FALSE

rtrimAllowsDate

Description

The rtrimAllowsDate property specifies whether the RTRIM database function allows a DATE/DATETIME parameter, and therefore whether the operation may be performed in the database or must be performed by the Campaign server.

Set the value to TRUE if the database is SQL Server, Oracle, or Teradata. The RTRIM function in these databases allows a DATE/DATETIME parameter.

Set the value to FALSE if the database is DB2. The RTRIM function in this database does not allow a DATE/DATETIME parameter.

Note that this setting is global, not per data source. If a value of no is recommended for any data source in use, set the value to no. If a value of yes is recommended for all data sources in use, set the value to yes.

Default value

TRUE

Valid Values

TRUE | FALSE

likeAllowsDate

Description

The likeAllowsDate property specifies whether the LIKE database function allows a DATE/DATETIME parameter, and therefore whether the operation may be performed in the database or must be performed by the Campaign server.

Set the value to TRUE if the database is SQL Server or Oracle. The LIKE function in these databases allows a DATE/DATETIME parameter.

Set the value to FALSE if the database is DB2 or Teradata. The LIKE function in these databases does not allow a DATE/DATETIME parameter.

Note: This setting is global, not per data source. If a value of no is recommended for any data source in use, set the value to no. If a value of yes is recommended for all data sources in use, set the value to yes.

Default value

TRUE

Valid Values

TRUE | FALSE

fileAllSpacesIsNull

Description

The fileAllSpacesIsNull property controls how Campaign interprets an empty field in a mapped flat file by specifying whether an all-spaces value in a flat file should be considered to be a NULL value.

When the value is TRUE, an all-spaces value is considered to be a NULL value. Campaign matches queries such as <field> is null, but fails queries such as <field> = "".

When the value is FALSE, an all-spaces value is treated as a non-NULL empty string. Campaign matches queries such as <field> = "", but fails <field> is null.

Default value

TRUE

Valid Values

TRUE | FALSE

Campaign | partitions | partition[n] | server | optimization

Properties in this category control IBM Campaign server optimization for each partition.

Note: This category is not related to IBM Contact Optimization.

maxVirtualMemory

Configuration category

`Campaign|partitions|partition[n]|server|optimization`

Description

This property specifies a default value for the maximum amount of system virtual memory to use when running a flowchart. You can increase the value to improve performance or decrease the value to limit the resources used by a single flowchart. The maximum value is 4095 MB. If you enter a larger value, Campaign automatically limits it to 4095 MB.

Set a value equal to (80% x available memory) / (number of expected concurrent flowcharts). For example:

If available virtual memory on server = 32 GB

Number of concurrent flowcharts = 10

Set virtual Memory = (80 % x 32) / 10 = approximately 2.5 GB / flowchart

Default value

128 (MB)

maxVirtualMemory is a global configuration setting. To override the value for a specific flowchart, open the flowchart in Edit mode, select **Advanced settings** from the **Admin** menu , select the **Server optimization** tab, and change the **IBM Campaign virtual memory usage** value.

useInDbOptimization

Configuration category

`Campaign|partitions|partition[n]|server|optimization`

Description

This property specifies whether IBM Campaign tries to perform as many operations as possible in the database instead of in the Campaign server.

Setting the value to `TRUE` can improve flowchart performance. When the value is `TRUE`, IBM Campaign avoids pulling the ID lists if possible.

When the value is `FALSE`, IBM Campaign maintains lists of IDs in the IBM Campaign server at all times.

Default value

`FALSE`

Valid Values

`TRUE | FALSE`

maxReuseThreads

Configuration category

Campaign | partitions | partition[n] | server | optimization

Description

This property specifies the number of operating system threads that are cached by the server process (unica_acsvr) for reuse. By default, the cache is disabled.

It is a best practice to use the cache when you want to reduce the overhead of thread allocation, or with operating systems that exhibit an inability to release threads when asked to do so by an application.

If the **maxReuseThreads** property is a non-zero value, set it to be greater than or equal to the value of **MaxQueryThreads**.

Default value

0 (zero), which disables the cache

threadStackSize

Configuration category

Campaign|partitions|partition[n]|server|optimization

Description

This property determines the number of bytes allocated for each thread's stack. Do not change this property except under guidance from IBM. The minimum value is 128 K. The maximum value is 8 MB.

Default value

1048576

tempTableDataSourcesForSegments

Configuration category

Campaign|partitions|partition[n]|server|optimization

Description

This property defines the list of data sources where persistent Segment temp tables can be created by the Create Seg process. This list is comma-separated. By default, this property is blank.

Default value

No default value defined.

doNotCreateServerBinFile

Configuration category

Campaign|partitions|partition[n]|server|optimization

Description

To improve performance, set this property to TRUE. When this property is TRUE, strategic segments create Segment temp tables in the data source rather than creating binary files on the IBM Campaign server. You must specify at least one data source in the Create Segment (CreateSeg) process configuration dialog to hold the temp tables. Also, you must set the AllowTempTables property to TRUE to enable the creation of temporary tables in a data source.

Default value

FALSE

Valid Values

TRUE | FALSE

forceViewForPreOptDates

Configuration category

Campaign|partitions|partition[n]|server|optimization

Description

The default value (TRUE) forces creation of a parameterized offer attribute view in a Mail List process whose offers are assigned from Optimize. A value of FALSE causes the parameterized offer attribute view to be created only if the Mail List exports at least one parameterized offer attribute.

If this value is set to FALSE, a Mail List process that is configured to get its input from an Extract process (whose source is an Optimize session) may write NULL values for EffectiveDate and ExpirationDate into the UA_Treatment table, even when the offer includes parameterized Effective and Expiration Dates. In this case, set it back to TRUE.

Default value

TRUE

Valid Values

TRUE | FALSE

httpCompressionForResponseLength

Configuration category

Campaign|partitions|partition[n]|server|optimization

Description

This property enables and configures compression for HTTP responses from the IBM Campaign web application to the client browser for flowchart-specific messages. The Campaign web application reads this property only once for each partition. If you modify this property, you must restart the web application for the change to take effect.

Compression can improve page load and interaction times by reducing the amount of data that is sent over HTTP.

All responses that have a data length greater than or equal to the httpCompressionForResponseLength value (in KB) are candidates for compression. Any other responses are not compressed.

Compression reduces network transfer, but it requires resources on the server side. Therefore, compression makes sense only for large amounts of data, when sufficient server-side resources are available. If you typically have network delays that can slow large data transfers, you can analyze how much time it takes to load a given amount of data. For example, suppose that some of your HTTP requests are <100 KB in size, but most are 300 to 500 KB. In this case, you would increase the value of this property to 500 KB so that only responses >= 500 KB in size are compressed.

To disable compression, set the value to 0.

Default value

100 (KB)

Valid Values

0 (disables compression) or higher

cacheSystemDSQueries

Configuration category

Campaign|partitions|partition[n]|server|optimization

Description

To improve performance, set this value to TRUE. When set to TRUE, this property reduces multiple execution of queries on the IBM Campaign system tables by caching the query results. When set to FALSE, query results are not cached.

Default value

TRUE

Valid Values

TRUE | FALSE

Campaign | partitions | partition[n] | server | logging

Properties in this category affect flowchart logging behavior for the specified partition on the IBM Campaign server.

enableWindowsEventLogging

Description

This property enables or disables IBM Campaign server logging to the Windows event log.

If the value is TRUE, logging to the Windows event log is enabled.

If the value is FALSE, logging to the Windows event log is disabled, and the **windowsEventLoggingLevel** and **windowsEventLoggingCategory** settings are ignored.

Attention: Windows Event logging can cause issues with flowchart runs. Avoid enabling this feature unless advised by Technical Support.

Default value

FALSE

Valid Values

TRUE | FALSE

logFileBufferSize

Configuration category

Campaign|partitions|partition[n]|server|logging

Description

This property is used when **keepFlowchartLogOpen** is TRUE. Specify a value to indicate the number of messages to buffer before writing to the

log. If the value is 1, every log message is written immediately to file, effectively disabling buffering but having a negative impact on performance.

This property is ignored if **keepFlowchartLogOpen** is FALSE.

Default value

5

keepFlowchartLogOpen

Configuration category

Campaign|partitions|partition[n]|server|logging

Description

This property specifies whether IBM Campaign opens and closes the flowchart log file each time a line is written to the log file.

A value of TRUE can improve performance of real-time interactive flowcharts. When the value is TRUE, IBM Campaign opens the flowchart log file only once, and closes it when the flowchart's server process exits. A side effect of using the TRUE value is that recently-logged messages may not be immediately visible in the log file, as IBM Campaign flushes the log messages to file only when its internal buffer becomes full or when the number of logged messages equals the value of the logFileBufferSize property.

If the value is FALSE, IBM Campaign opens and closes the flowchart log file.

Default value

FALSE

Valid Values

TRUE | FALSE

logProcessId

Configuration category

Campaign|partitions|partition[n]|server|logging

Description

This property controls whether the process ID (pid) of the IBM Campaign Server process is included in the log file.

If the value is TRUE, the process ID is logged.

If the value is FALSE, the process ID is not logged.

Default value

TRUE

Valid Values

TRUE | FALSE

logMaxBackupIndex

Configuration category

Campaign|partitions|partition[n]|server|logging

Description

This property specifies the number of backup Campaign server log files that are kept before the oldest is erased.

If the value is `0` (zero), no backup files are created, and the log file is truncated when it reaches the size specified by the `logFileMaxSize` property.

For a value of `n`, where `n` is greater than zero, the files {`File.1`, ..., `File.n-1`} are renamed to {`File.2`, ..., `File.n`}. Also, `File` is renamed `File.1` and closed. A new `File` is created to receive further log output.

Default value

1 (creates one backup log file)

loggingCategories

Configuration category

`Campaign|partitions|partition[n]|server|logging`

Description

This property specifies the category of messages written to the IBM Campaign server flowchart log file. This property works in conjunction with **loggingLevels**, which determines the severity of messages to log for all selected categories.

Specify one or more categories in a comma-separated list. Use `ALL` as shorthand to indicate that you want to log all categories.

The values that you specify determine which events are logged by default for all flowcharts. Users can override the default selections by opening a flowchart for editing and choosing **Options** > **Logging options**. The corresponding Logging Options are indicated below in parentheses, after each configuration value.

Default value

`ALL`

Valid Values

`ALL`
`BAD_ORDER` (Log ID ordering errors)
`CELL_ACCESS` (Cell level operations)
`CONFIG` (Log config settings at start of run)
`DATA_ERRORS` (Log data conversion errors)
`DBLOAD` (External DB loader operations)
`FILE_ACCESS`(File operations)
`GENERAL` (Others)
`COMMANDS` (External interface)
`MEMORY` (Memory allocation)
`PROCRUN` (Process Run)
`QUERY` (Queries issues against user tables)
`SORT` (Log data sorting progress)
`SYSQUERY` (Queries issued against system tables)
`TABLE_ACCESS` (Table level operations)

TABLE_MAPPING (Log table mapping info at start of run)

TABLE_IO (Log data I/O process)

WEBPROC (Web server interface)

loggingLevels

Configuration category

Campaign|partitions|partition[n]|server|logging

Description

The **loggingLevels** property controls the amount of detail written to the Campaign server log file, based on severity.

Default value

MEDIUM

Valid Values

LOW: represents the least detail (the most severe errors only)

MEDIUM

HIGH

ALL: includes trace messages and is intended primarily for diagnostic purposes

Note: You may want to set **loggingLevels** to ALL during configuration and testing. This value generates a large amount of data and therefore may not be advisable for production operation. Setting any logging level higher than its default can adversely affect performance.

You can adjust these settings from within a flowchart by using **Tools** > **Logging options**.

windowsEventLoggingCategories

Configuration category

Campaign|partitions|partition[n]|server|logging

Description

This property specifies the category of messages written to the Windows event log for the Campaign server. This property works in conjunction with **windowsEventLoggingLevels**, which determines which messages are logged based on severity (for all selected categories).

You can specify multiple categories in a comma-separated list. The category all provides a shorthand for specifying all logging categories.

Default value

ALL

Valid Values

ALL

BAD_ORDER

CELL_ACCESS

CONFIG

DATA_ERRORS

DBLOAD

FILE_ACCESS

GENERAL

COMMANDS

MEMORY

PROCRUN

QUERY

SORT

SYSQUERY

TABLE_ACCESS

TABLE_MAPPING

TABLE_IO

WEBPROC

logFileMaxSize

Configuration category

Campaign|partitions|partition[n]|server|logging

Description

This property specifies the maximum size, in bytes, that the Campaign server log file is allowed to reach before being rolled over to backup files.

Default value

10485760 (10 MB)

windowsEventLoggingLevels

Configuration category

Campaign|partitions|partition[n]|server|logging

Description

This property controls the amount of detail written to the Windows event log for the IBM Campaign server, based on severity.

Default value

MEDIUM

Valid Values

LOW: represents the least detail (the most severe errors only)

MEDIUM

HIGH

ALL: includes trace messages and is intended for diagnostic purposes.

enableLogging

Configuration category

Campaign|partitions|partition[n]|server|logging

Description

This property specifies whether IBM Campaign server logging is turned on at session startup.

If the value is TRUE, logging is turned on.

If the value is FALSE, logging is turned off.

Default value

TRUE

Valid Values

TRUE | FALSE

AllowCustomLogPath

Configuration category

Campaign|partitions|partition[n]|server|logging

Description

This property allows users to change the log path for each flowchart that generates flowchart-specific logging information when it is run. By default, all flowchart log files are saved in Campaign_home/partitions/partition_name/logs.

A setting of TRUE allows users to change the path through the user interface or when using unica_svradm to run the flowchart.

A setting of FALSE prevents users from changing the path to which the flowchart log file is written.

Default value

FALSE

Valid Values

TRUE | FALSE

Campaign | partitions | partition[n] | server | flowchartRun

Properties in this category specify how many errors are allowed in a Campaign Snapshot export, what files are saved when you save a flowchart, and the maximum number of IDs for each top-level process in a test run.

maxDataErrorsAllowed

Description

The maxDataErrorsAllowed property specifies the maximum number of data conversion errors allowed in an Campaign Snapshot export.

Default value

0 (zero), which allows no errors

saveRunResults

Description

This property lets you save Campaign flowchart run results in the temporary folder and database temp tables. You can adjust this option for individual flowcharts by using **Admin** > **Advanced Settings** when editing a flowchart.

For flowcharts that create artifacts that you want to save, you must set **saveRunResults** to TRUE. For example, if you have flowcharts that include **CreateSeg** processes, you must save run results. If you do not save run results, strategic segments do not persist.

When the value is TRUE, the flowchart ("underscore") files are saved and, if you are using **useInDbOptimization**, database temp tables persist.

When the value is FALSE, only the .ses file is saved. Therefore, you cannot view intermediate results if you reload the flowchart.

IBM Campaign creates many temporary files in the temporary directory, which can cause file systems to be highly utilized or even full. Setting this property to FALSE makes the flowchart clean up those files after run completion. However, using a setting of FALSE is not always feasible because it prevents you from doing partial flowchart runs.

To save disk space, you can create your own script to delete files in the temporary folder, but you must never delete files for flowcharts that are currently running. To avoid flowchart failures, never delete any files from the temporary folder that are being updated or created today. For maintenance purposes, you can delete files from the temp folder if they are older than 2 days.

Default value

TRUE

Valid Values

TRUE | FALSE

testRunDefaultSize

Description

The testRunDefaultSize property specifies the default maximum number of IDs for each top-level process in a Campaign test run. A value of 0 (zero) removes the limitation on the number of IDs.

Default value

0 (zero)

Campaign | partitions | partition[n] | server | profile

Properties in this category specify the maximum number of categories that are created during profiling for numeric and text values in Campaign.

profileMaxTextCategories

Description

The profileMaxTextCategories and profileMaxNumberCategories properties specify the maximum number of categories created in Campaign during profiling for text and numeric values, respectively.

These values are different from the setting for the number of bins displayed to the user, which can be modified through the user interface.

Default value

1048576

profileMaxNumberCategories

Description

The profileMaxNumberCategories and profileMaxTextCategories properties specify the maximum number of categories created in Campaign during profiling for numeric and text values, respectively.

These values are different from the setting for the number of bins displayed to the user, which can be modified through the user interface.

Default value

1024

Campaign | partitions | partition[n] | server | internal

Properties in this category specify integration settings and the internalID limits for the selected Campaign partition. If your Campaign installation has multiple partitions, set these properties for each partition that you want to affect.

internalIdLowerLimit

Configuration category

Campaign|partitions|partition[n]|server|internal

Description

The internalIdUpperLimit and internalIdLowerLimit properties constrain the Campaign internal IDs to be within the specified range. Note that the values are inclusive: that is, Campaign may use both the lower and upper limit.

Default value

0 (zero)

internalIdUpperLimit

Configuration category

Campaign|partitions|partition[n]|server|internal

Description

The internalIdUpperLimit and internalIdLowerLimit properties constrain the Campaign internal IDs to be within the specified range. The values are inclusive: that is, Campaign may use both the lower and upper limit. If Distributed Marketing is installed, set the value to 2147483647.

Default value

4294967295

eMessageInstalled

Configuration category

Campaign|partitions|partition[n]|server|internal

Description

Indicates that eMessage is installed. When you select Yes, eMessage features are available in the Campaign interface.

The IBM installer sets this property to Yes for the default partition in your eMessage installation. For additional partitions where you installed eMessage, you must configure this property manually.

Default value

No

Valid Values

Yes | No

interactInstalled

Configuration category

Campaign|partitions|partition[n]|server|internal

Description

After installing the Interact design environment, this configuration property should be set to Yes to enable the Interact design environment in Campaign.

If Interact is not installed, set to No. Setting this property to No does not remove Interact menus and options from the user interface. To remove menus and options, you must manually unregister Interact using the configTool utility.

Default value

No

Valid Values

Yes | No

Availability

This property is applicable only if you installed Interact.

MO_UC_integration

Configuration category

Campaign|partitions|partition[n]|server|internal

Description

Enables integration with Marketing Operations for this partition, if the integration is enabled in the **Platform** configuration settings. For more information, see the *IBM Marketing Operations and Campaign Integration Guide*.

Default value

No

Valid Values

Yes | No

MO_UC_BottomUpTargetCells

Configuration category

Campaign|partitions|partition[n]|server|internal

Description

For this partition, allows bottom-up cells for Target Cell Spreadsheets, if **MO_UC_integration** is enabled. When set to Yes, both top-down and bottom-up target cells are visible, but bottom-up target cells are read-only. For more information, see the *IBM Marketing Operations and Campaign Integration Guide*.

Default value

No

Valid Values

Yes | No

Legacy_campaigns

Configuration category

Campaign|partitions|partition[n]|server|internal

Description

For this partition, enables access to campaigns created before Marketing Operations and Campaign were integrated. Applies only if **MO_UC_integration** is set to Yes. Legacy campaigns also include campaigns created in Campaign 7.x and linked to Plan 7.x projects. For more information, see the *IBM Marketing Operations and Campaign Integration Guide*.

Default value

No

Valid Values

Yes | No

IBM Marketing Operations - Offer integration

Configuration category

Campaign|partitions|partition[n]|server|internal

Description

Enables the ability to use Marketing Operations to perform offer lifecycle management tasks on this partition, if **MO_UC_integration** is enabled for this partition. Offer integration must be enabled in your **Platform** configuration settings. For more information, see the *IBM Marketing Operations and Campaign Integration Guide*.

Default value

No

Valid Values

Yes | No

UC_CM_integration

Configuration category

Campaign|partitions|partition[n]|server|internal

Description

Enables Digital Analytics online segment integration for a Campaign partition. If you set this value to Yes, the Select process box in a flowchart provides the option to select **Digital Analytics Segments** as input. To configure the Digital Analytics integration for each partition, choose **Settings > Configuration > Campaign | partitions | partition[n] | Coremetrics**.

Default value

No

Valid Values

Yes | No

numRowsReadToParseDelimitedFile

Configuration category

Campaign|partitions|partition[n]|server|internal

Description

This property is used when mapping a delimited file as a user table. It is also used by the Score process box when importing a score output file from IBM SPSS Modeler Advantage Enterprise Marketing Management Edition. To import or map a delimited file, Campaign needs to parse the file to identify the columns, data types (field types), and column widths (field lengths).

The default value of 100 means Campaign examines the first 50 and the last 50 line entries in the delimited file. Campaign then allocates the field length based on the largest value it finds within those entries. In most cases, the default value is sufficient to determine field lengths. However, in very large delimited files, a later field might exceed the estimated length that Campaign computes, which can cause an error during flowchart runtime. Therefore, if you are mapping a very large file, you can increase this value to make Campaign examine more line entries. For example, a value of 200 makes Campaign examine the first 100 line entries and the last 100 line entries of the file.

A value of 0 examines the entire file. Typically, this is necessary only if you are importing or mapping files that have variable data widths of fields which cannot be identified by reading the first and last few lines. Reading the entire file for extremely large files can increase the required processing time for table mapping and Score process box runs.

Default value

100

Valid Values

0 (all lines) or any positive integer

Campaign | partitions | partition[n] | server | fileDialog

Properties in this category specify the default directories for Campaign input and output data files.

defaultOutputDirectory

Description

The defaultOutputDirectory property specifies the path used to initialize the Campaign File Selection dialog. The defaultOutputDirectory property is used when an output data file is mapped into Campaign. If no value is specified, the path is read from the environment variable UNICA_ACDFDIR.

Default value

No default value defined.

defaultInputDirectory

Description

The defaultInputDirectory property specifies the path used to initialize the Campaign File Selection dialog. The defaultInputDirectory property is

used when an input data file is mapped into Campaign. If no value is specified, the path is read from the environment variable UNICA_ACDFDIR.

Default value

No default value defined.

Campaign | partitions | partition[n] | offerCodeGenerator

Properties in this category specify the class, classpath, and configuration string for the offer code generator, and also the cell code generator used to assign a contact process to a Target Cell Spreadsheet cell.

offerCodeGeneratorClass

Description

The offerCodeGeneratorClass property specifies the name of the class Campaign uses as its offer code generator. The class must be fully qualified with its package name.

Default value

Note that line breaks have been added for print.

```
com.unica.campaign.core.codegenerator.samples.
ExecutableCodeGenerator
```

offerCodeGeneratorConfigString

Description

The offerCodeGeneratorConfigString property specifies a string that is passed into the offer code generator plug-in when it is loaded by Campaign. By default, the ExecutableCodeGenerator (shipped with Campaign) uses this property to indicate the path (relative to Campaign application home directory) to the executable to run.

Default value

```
./bin
```

defaultGenerator

Description

The defaultGenerator property specifies the generator for the cell codes that appear in contact-style process boxes and are used to assign cells to Target Control Spreadsheet cells. The Target Control Spreadsheet manages cell and offer mappings for campaigns and flowcharts.

Default value

```
uacoffercodegen.exe
```

offerCodeGeneratorClasspath

Description

The offerCodeGeneratorClasspath property specifies the path to the class Campaign uses as its offer code generator. It can be either a full path or a relative path.

If the path ends in a slash (forward slash / for UNIX or backslash \ for Windows), Campaign assumes it to be a path to a directory that contains

the Java plug-in class that should be used. If the path does not end in a slash, Campaign assumes it is the name of a `jar` file that contains the Java class.

If the path is relative, Campaign assumes it is relative to the Campaign application home directory.

Default value

`codeGenerator.jar` (packaged in the `Campaign.war` file)

Campaign | monitoring

Properties in the this category specify whether the Operational Monitoring feature is enabled, the URL of the Operational Monitoring server, and caching behavior. Operational Monitoring displays and allows you to control active flowcharts.

cacheCleanupInterval

Description

The `cacheCleanupInterval` property specifies the interval, in seconds, between automatic cleanups of the flowchart status cache.

This property is not available in versions of Campaign earlier than 7.0.

Default value

`600` (10 minutes)

cacheRunCompleteTime

Description

The `cacheRunCompleteTime` property specifies the amount of time, in minutes, that completed runs are cached and display on the Monitoring page.

This property is not available in versions of Campaign earlier than 7.0.

Default value

`4320`

monitorEnabled

Description

The `monitorEnabled` property specifies whether the monitor is turned on.

This property is not available in versions of Campaign earlier than 7.0.

Default value

`FALSE`

Valid values

`TRUE | FALSE`

serverURL

Description

The `Campaign > monitoring > serverURL` property specifies the URL of the Operational Monitoring server. This is a mandatory setting; modify the value if the Operational Monitoring server URL is not the default.

If Campaign is configured to use Secure Sockets Layer (SSL) communications, set the value of this property to use HTTPS. For example: `serverURL=https://host:SSL_port/Campaign/OperationMonitor` where:

- *host* is the name or IP address of the machine on which the web application is installed
- *SSL_Port* is the SSL port of the web application.

Note the `https` in the URL.

Default value

`http://localhost:7001/Campaign/OperationMonitor`

monitorEnabledForInteract

Description

If set to `TRUE`, enables Campaign JMX connector server for Interact. Campaign has no JMX security.

If set to `FALSE`, you cannot connect to the Campaign JMX connector server.

This JMX monitoring is for the Interact contact and response history module only.

Default value

`FALSE`

Valid Values

`TRUE | FALSE`

Availability

This property is applicable only if you have installed Interact.

protocol

Description

Listening protocol for the Campaign JMX connector server, if `monitorEnabledForInteract` is set to yes.

This JMX monitoring is for the Interact contact and response history module only.

Default value

JMXMP

Valid Values

JMXMP | RMI

Availability

This property is applicable only if you have installed Interact.

port

Description

Listening port for the Campaign JMX connector server, if `monitorEnabledForInteract` is set to yes.

This JMX monitoring is for the Interact contact and response history module only.

Default value

2004

Valid Values

An integer between 1025 and 65535.

Availability

This property is applicable only if you have installed Interact.

Campaign | ProductReindex

The creator of an offer can specify the products that are associated with that offer. When the list of products available for association with offers changes, the offer/product associations must be updated. Properties in the Campaign > ProductReindex category specify the frequency of these updates and the time of day that the first update runs.

startTime

Description

The `startTime` property specifies the time of day when offer/product associations are updated for the first time. The first update occurs on the day after the Campaign server is started, and subsequent updates occur at intervals specified in the `interval` parameter. The format is `HH:mm:ss`, using a 24-hour clock.

Note that when Campaign first starts up, the `startTime`property is used according to the following rules:

- If the time of day specified by `startTime` is in the future, the first offer/product associations update will occur at `startTime` of the current day.
- If `startTime` is in the past for the current day, the first update will occur at `startTime` tomorrow, or at `interval` minutes from the current time, whichever is earlier.

Default value

`12:00:00` (noon)

interval

Description

The `interval` property specifies the time, in minutes, between updates of offer/product associations. The update occurs for the first time at the time specified in the `startTime` parameter, on the day after the Campaign server is started.

Default value

`3600` (60 hours)

Campaign | unicaACListener

If you are configuring a single-node listener cluster, use this category, and only this category, to define configuration settings for your non-clustered listener. For clustered listeners, the properties in this category pertain to all of the listener nodes in the cluster, except for the following properties, which are ignored:

serverHost, serverPort, useSSLForPort2, serverPort2. (Instead, set these properties for each individual node, under Campaign|unicaACListener|node[n].)

These properties must be set only once per instance of Campaign; they do not need to be set for every partition.

enableWindowsImpersonation

Configuration category

Campaign|unicaACListener

Description

This property applies to both clustered and non-clustered listener configurations. For a clustered configuration, this property applies to all of the listener nodes in the cluster.

The enableWindowsImpersonation property specifies whether Windows impersonation is enabled in Campaign.

Set the value to TRUE if you are using Windows impersonation. You must configure Windows impersonation separately if you want to leverage the Windows-level security permissions for file access.

Set the value to FALSE if you are not using Windows impersonation.

Default value

FALSE

Valid Values

TRUE | FALSE

enableWindowsEventLogging

Configuration category

Campaign|unicaACListener

Description

This property applies to both clustered and non-clustered listener configurations. For a clustered configuration, this property applies to all of the listener nodes in the cluster.

The Campaign | unicaACListener | enableWindowsEventLogging property turns Windows event logging on or off for IBM Campaign listener events. Set this property to TRUE to log to the Windows event log.

Attention: Windows Event logging can cause issues with flowchart runs. Avoid enabling this feature unless advised by Technical Support.

Default value

FALSE

Valid Values

TRUE | FALSE

serverHost

Configuration category

Campaign|unicaACListener

Description

If you have a single-node listener configuration, this property identifies the listener. If you have a clustered listener configuration, this property is ignored. (Instead, set this property for each individual node, under Campaign|unicaACListener|node[n].)

The serverHost property specifies the name or IP address of the machine where the Campaign listener is installed. If the Campaign listener is not installed on the same machine where IBM Marketing Software is installed, change the value to the machine name or IP address of the machine where the Campaign listener is installed.

Default value

localhost

logMaxBackupIndex

Configuration category

Campaign|unicaACListener

Description

This property applies to both clustered and non-clustered listener configurations. For a clustered configuration, this property applies to all of the listener nodes in the cluster.

The logMaxBackupIndex property specifies how many backup files can exist before the oldest one is deleted. If you set this property to 0 (zero), Campaign does not create any backup files and the log file stops logging when it reaches the size you specified in the logMaxFileSize property.

If you specify a number (N) for this property, when the log file (File) reaches the size you specified in the logMaxFileSize property, Campaign renames the existing backup files (File.1 ... File.N-1) to File.2 ... File.N , renames the current log file File.1, closes it, and and starts a new log file named File.

Default value

1 (creates one backup file)

logStringEncoding

Configuration category

Campaign|unicaACListener

Description

This property applies to both clustered and non-clustered listener configurations. For a clustered configuration, this property applies to all of the listener nodes in the cluster.

The logStringEncoding property controls the encoding used for all log files. This value must match the encoding used on the operating system. For multi-locale environments, UTF-8 is the preferred setting.

If you change this value, you should empty or remove all affected log files to prevent writing multiple encodings into a single file.

Note: WIDEUTF-8 is not supported for this setting.

Default value

native

Valid Values

See "Character encodings in Campaign" in the *Campaign Administrator's Guide*.

systemStringEncoding

Configuration category

Campaign|unicaACListener

Description

This property applies to both clustered and non-clustered listener configurations. For a clustered configuration, this property applies to all of the listener nodes in the cluster.

The systemStringEncoding property indicates which encodings Campaign uses to interpret values received from and sent to the operating system, such as file system paths and filenames. In most cases, you can set this value to native. For multi-locale environments, use UTF-8.

You can specify more than one encoding, separated by commas. For example:

UTF-8,ISO-8859,CP950

Note: WIDEUTF-8 is not supported for this setting.

Default value

native

Valid Values

See *Character encodings in Campaign* in the *Campaign Administrator's Guide*.

loggingLevels

Configuration category

Campaign|unicaACListener

Description

This property applies to both clustered and non-clustered listener configurations. For a clustered configuration, this property applies to all of the listener nodes in the cluster.

The Campaign > unicaACListener > loggingLevels property controls the amount of detail written to the log file.

This property applies to both clustered and non-clustered configurations.

Default value

MEDIUM

Valid Values

- LOW
- MEDIUM
- HIGH

maxReuseThreads

Configuration category

Campaign|unicaACListener

Description

This property applies to both clustered and non-clustered listener configurations. For a clustered configuration, this property applies to all of the listener nodes in the cluster.

This property sets the number of operating system threads cached by the Campaign listener process (`unica_aclsnr`) for reuse.

It is a best practice to use the cache when you want to reduce the overhead of thread allocation, or with operating systems that can exhibit an inability to release threads when asked to do so by an application.

This property applies to both clustered and non-clustered configurations.

Default value

`0` (zero), which disables the cache

logMaxFileSize

Configuration category

`Campaign|unicaACListener`

Description

This property applies to both clustered and non-clustered listener configurations. For a clustered configuration, this property applies to all of the listener nodes in the cluster.

The `logMaxFileSize` property specifies the maximum size, in bytes, that the log file can reach before rolling into the backup file.

This property applies to both clustered and non-clustered configurations.

Default value

`10485760` (10 MB)

windowsEventLoggingLevels

Configuration category

`Campaign|unicaACListener`

Description

This property applies to both clustered and non-clustered listener configurations. For a clustered configuration, this property applies to all of the listener nodes in the cluster.

The `windowsEventLoggingLevels` property controls the amount of detail written to the Windows event log file based on severity.

This property applies to both clustered and non-clustered configurations.

Default value

`MEDIUM`

Valid Values

- `LOW`
- `MEDIUM`
- `HIGH`
- `ALL`

The `ALL` level includes trace messages intended for diagnostic purposes.

serverPort

Configuration category

Campaign|unicaACListener

Description

If you have a single-node listener configuration, this property identifies the listener port. If you have a clustered listener configuration, this property is ignored. (Instead, set this property for each individual node, under Campaign | unicaACListener | node[n].)

The serverPort property specifies the port where a single (non-clustered) Campaign listener is installed.

Default value

4664

useSSL

Configuration category

Campaign|unicaACListener

Description

This property applies to both clustered and non-clustered listener configurations. For a clustered configuration, this property applies to all of the listener nodes in the cluster.

The useSSL property specifies whether to use Secure Sockets Layer for communications between the Campaign listener and the Campaign web application.

Also see the description for the serverPort2 property, in this category.

Default value

no

Valid Values

yes | no

serverPort2

Configuration category

Campaign|unicaACListener

Description

This property is optional.

This property applies only to a single-node listener configuration. If you have a clustered listener configuration, this property is ignored. (Instead, define serverPort2 for each individual node, under Campaign | unicaACListener | node[n].)

The serverPort2 property, in conjunction with the useSSLForPort2 property, also in this category, enables you to specify the use of SSL for communication between the Campaign listener and flowchart processes, separately from the communication between the Campaign web application and listener, which is specified by the serverPort and useSSL properties in this category.

All communication between Campaign components, (between the web application and listener and between the listener and server) use the mode specified by the useSSL property under any of the following conditions.

- serverPort2 is set to its default value of 0, **or**
- serverPort2 is set to the same value as serverPort, **or**
- useSSLForPort2 is set to the same value as useSSL

In these cases, a second listener port is not enabled, and communication between the Campaign listener and the flowchart (server) processes and communication between the listener and the Campaign web application use the same mode: either both non-SSL or both SSL, depending on the value of the useSSL property.

The listener uses two different modes of communication when both of the following conditions exist.

- serverPort2 is set to a non-0 value different from the value of serverPort, **and**
- useSSLForPort2 is set to a value different from the value of useSSL

In this case, a second listener port is enabled, and the listener and flowchart processes use the mode of communication specified by useSSLForPort2.

The Campaign web application always uses the communication mode specified by useSSL when communicating to the listener.

When SSL is enabled for communication between the Campaign listener and flowchart processes, set the value of this property (serverPort2) to an appropriate port.

Default value

0

useSSLForPort2

Configuration category

Campaign|unicaACListener

Description

This property applies only to a single-node listener configuration. If you have a clustered listener configuration, this property is ignored. (Instead, define useSSLForPort2 for each individual node, under Campaign | unicaACListener | node[n].)

For information, see the description for serverPort2, in this category.

Default value

FALSE

Valid Values

TRUE|FALSE

keepalive

Configuration category

Campaign|unicaACListener

Description

This property applies to both clustered and non-clustered listener configurations. For a clustered configuration, this property applies to all of the listener nodes in the cluster.

Use the keepalive property to specify, in seconds, the frequency with which the Campaign web application server sends keep alive messages on otherwise-inactive socket connections to the Campaign listener.

Using the keepalive configuration parameter enables socket connections to remain open through extended periods of application inactivity in environments configured to close inactive connections between the web application and the listener (for example, a firewall).

When there is activity on a socket, the keep alive period is automatically reset. At the DEBUG logging level in the web application server, the campaignweb.log will show evidence of the keep alive messages as they are sent to the listener.

Default value

0, which disables the keepalive feature

Valid Values

positive integers

loggingCategories

Configuration category

Campaign|unicaACListener

Description

This property specifies the category of messages written to the Campaign listener log file.

The loggingCategories property in conjunction with the loggingLevels property determines the severity of the messages that are logged for all selected categories.

Specify one or more categories in a comma separated list. Use the ALL option to include logs for all categories.

Default value

ALL

Valid Values

Note: The corresponding Logging Options are indicated in parentheses after each configuration value.

ALL

GENERAL (Others)

COMMANDS (External interface)

SYS CALL (System Call)

UDB (udb)

XML (xml)

Campaign | unicaACListener | node [n]

A non-clustered listener configuration should not have any nodes under this category. Nodes are created and used only for clustered listener configurations. For a clustered listener configuration, configure an individual child node for each listener in the cluster.

If clustering is enabled, you must configure at least one child node or an error occurs during startup.

Important: Never remove a node from the configuration unless you first stop all of the clustered listener nodes. Otherwise, any existing sessions on the removed listener continue to run but the master listener will not be able to contact the removed listener node. This can cause unexpected results.

serverHost

Configuration category

`Campaign|unicaACListener|node[n]`

Description

This property applies only if you have a clustered listener configuration. This property identifies each individual listener node in a cluster.

For each node, specify the hostname of the machine where the Campaign listener is installed.

Default value

No default value assigned.

serverPort

Configuration category

`Campaign|unicaACListener|node[n]`

Description

This property applies only if you have a clustered listener configuration. This property identifies the port that is used for communication between each clustered listener node and the IBM Campaign web application server.

The specified port is also used for communication between listener nodes.

Default value

No default value assigned.

useSSLForPort2

Configuration category

`Campaign|unicaACListener|node[n]`

Description

Optional. This property applies only if you have a clustered listener configuration. You can set this property for each clustered listener node. For information about how to use this property, read the description for `Campaign|unicaACListener|serverPort2`.

Default value

`FALSE`

Valid Values

TRUE|FALSE

serverPort2

Configuration category

Campaign|unicaACListener|node[n]

Description

Optional. This property applies only if you have a clustered listener configuration. You can set this property for each clustered listener node. For information about how to use this property, read the description for Campaign|unicaACListener|serverPort2.

Default value

3

masterListenerPriority

Configuration category

Campaign|unicaACListener|node[n]

Description

This property applies only if you have a clustered listener configuration.

A cluster always includes one master listener. All clients, including the IBM Campaign web server application, the Campaign Server Manager (unica_svradm), and utilities such as unica_acsesutil, use the masterListenerPriority to identify the master listener.

Any node in the cluster can act as the master listener. The masterListenerPriority determines which node initially acts as the master listener. It also determines which listener will take over as the master listener in failover situations. Ideally, the listener node with the most processing power should be assigned the highest priority.

Priority 1 is the highest priority. Assign a 1 to the machine that you want to be the master listener. That machine will serve as the master listener unless it goes down or cannot be contacted, for example due to a network issue. Assign a 2 to the next machine, and so on.

You must assign a priority to every listener in the cluster. If you do not want a machine to serve as the master listener, assign it the lowest priority (10). However, you cannot prohibit a listener from being designated as master. In a clustered listener configuration, one listener must always serve as the master.

If the designated master listener cannot be contacted, then the next machine becomes the master listener, based on its assigned priority.

If multiple nodes have the same priority, the system selects the first of those nodes from the list of nodes in this category.

Note: After you change the priority, run the unica_svradm refresh command to inform the master listener of the change.

Default value

No default value assigned.

Valid Values

1 (high) through 10 (low)

loadBalanceWeight

Configuration category

Campaign|unicaACListener|node[n]

Description

This property applies only if you have a clustered listener configuration. This property controls load balancing among clustered nodes. Each node in a cluster can process a portion of the total application traffic. Adjust the weight for each listener node to determine how much load the node will get. A higher value assigns a greater proportion of the load, so more transactions are given to that listener node.

Assign higher values to machines that have more processing capacity. Assign lower values to less powerful or more heavily loaded machines. A value of 0 prohibits the listener from processing any transactions and typically is not used. If multiple nodes have the same weight, the system selects the first of those nodes from the list of nodes in this category.

Note: After you change the weight, run the `unica_svradm refresh` command to inform the master listener of the change.

Example

You have three physical hosts: A, B, and C. Host A is the most powerful machine and Host C is the least capable, so you assign weights as follows: A=4, B=3, C=2. To see how requests will be allocated, sum the weights and divide by the number of listeners: 4+3+2=9. In this scenario, listener A processes 4 of every 9 transactions, listener B processes 3 of every 9 transactions, and listener C processes the remaining 2 of every 9 transactions. The scheduling sequence for this cluster is **AA**B**A**BC**A**BC. As requests come in, the sequence continues, ensuring that the requests are distributed across the nodes.

Default value

No default value assigned.

Valid Values

0 through 10 (highest priority)

Campaign | campaignClustering

Set these properties if you have a clustered listener configuration. Set these properties once per instance of Campaign; you do not need to set them for every partition.

enableClustering

Configuration category

Campaign|campaignClustering

Description

If you have a single listener, leave the value set to `FALSE`. This causes all of the other properties in this category to be ignored, because they do not apply to a single-node configuration.

For a clustered listener configuration, set the value to TRUE, configure the other properties in this category, then configure listener nodes under Campaign|unicaACListener|node[n]. When the value is TRUE, you must define at least one child node. If you do not define at least one child node, an error occurs during startup.

When the value is TRUE, the following properties are ignored for Campaign|unicaACListener and are instead defined for each individual node under Campaign|unicaACListener|node[n]: serverHost, serverPort, serverPort2, useSSLForPort2.

Default value

FALSE

Valid Values

TRUE|FALSE

masterListenerLoggingLevel

Configuration category

Campaign|campaignClustering

Description

This property applies only if enableClustering is TRUE. This property controls the amount of detail that is written to the master listener log file (<campaignSharedHome>/logs/masterlistener.log).

The default value of LOW provides the least detail (only the most severe error messages are written). ALL includes trace debug messages and is intended for diagnostic purposes.

Default value

MEDIUM

Valid Values

LOW|MEDIUM|HIGH|ALL

masterListenerHeartbeatInterval

Configuration category

Campaign|campaignClustering

Description

This property applies only if enableClustering is TRUE. This property affects the master listener. Specify how often the master listener tries to connect to all of the configured listener nodes to identify their availability. When the master listener connects to the nodes for availability, it also sends a heartbeat message to inform them that the master listener is alive. Therefore, this property serves two purposes (1) Heartbeat from master listener (2) Status response from each listener node.

Default value

10 seconds

webServerDelayBetweenRetries

Configuration category

Campaign|campaignClustering

Description

This property applies only if enableClustering is TRUE. This property specifies the time delay between retries for the IBM Campaign web application server to attempt to connect to the IBM Campaign listener.

Default value

5 seconds

webServerRetryAttempts

Configuration category

Campaign|campaignClustering

Description

This property applies only if enableClustering is TRUE. This property indicates how many times the IBM Campaign web application server attempts to connect to the IBM Campaign listener.

Default value

3

campaignSharedHome

Configuration category

Campaign|campaignClustering

Description

This property applies only if enableClustering is TRUE.

In a clustered configuration, listener nodes share the files and folders indicated below. The shared location is specified at installation time.

```
campaignSharedHome
 |--->/conf
    |------> activeSessions.udb
    |------> deadSessions.udb
    |------> etc.
 |--->/logs
    |------> masterlistener.log
    |------> etc.
 |--->/partitions
    |------> partition[n]
       |----------> {similar to <Campaign_home> partition folder structure}
```

Note: Each listener also has its own set of folders and files that are not shared, at <Campaign_home> (the IBM Campaign application installation directory).

masterListenerloggingCategories

Configuration category

Campaign|campaignClustering

Description

This property specifies the category of messages written to the Campaign master listener log file.

The masterListenerloggingCategories property in conjunction with the masterListenerLoggingLevel property determines the severity of the messages that are logged for all selected categories.

Specify one or more categories in a comma separated list. Use the ALL option to include logs for all categories.

Default value

ALL

Valid Values

Note: The corresponding Logging Options are indicated in parentheses after each configuration value.

ALL

FILE_ACCESS (File operations)

GENERAL (Others)

COMMANDS (External interface)

Campaign | unicaACOOptAdmin

These configuration properties define settings for the unicaACOOptAdmin tool.

getProgressCmd

Description

Specifies a value that is used internally. Do not change this value.

Default value

optimize/ext_optimizeSessionProgress.do

Valid Values

optimize/ext_optimizeSessionProgress.do

runSessionCmd

Description

Specifies a value that is used internally. Do not change this value.

Default value

optimize/ext_runOptimizeSession.do

Valid Values

optimize/ext_runOptimizeSession.do

loggingLevels

Description

The loggingLevels property controls the amount of detail that is written to the log file for the Contact Optimization command-line tool, which is based on severity. Available levels are LOW, MEDIUM, HIGH, and ALL, with LOW providing the least detail (that is, only the most severe messages are written). The ALL level includes trace messages and is intended primarily for diagnostic purposes.

Default value

HIGH

Valid Values

LOW | MEDIUM | HIGH | ALL

cancelSessionCmd

Description

Specifies a value that is used internally. Do not change this value.

Default value

optimize/ext_stopOptimizeSessionRun.do

Valid Values

optimize/ext_stopOptimizeSessionRun.do

logoutCmd

Description

Specifies a value that is used internally. Do not change this value.

Default value

optimize/ext_doLogout.do

Valid Values

optimize/ext_doLogout.do

getProgressWaitMS

Description

Set this value to the number (integer) of milliseconds between two successive polls to the web application to get progress information. This value is not used if you do not set `getProgressCmd`.

Default value

1000

Valid Values

An integer greater than zero

Campaign | server

The property in this category specifies a URL that is used internally, and does not need to be changed.

fullContextPath

Description

The `fullContextPath` specifies the URL that Campaign flowcharts use to communicate to the application server Listener proxy. This property is undefined by default, which causes the system to determine the URL dynamically. When IBM Marketing Platform is integrated with the IBM Tivoli® web access control platform, you must set this property to the Campaign URL in Tivoli.

Default value

No default value defined.

numRetryServerCommand

Description

The `numRetryServerCommand` specifies the maximum number of times the IBM Campaign web application can call the IBM Campaign analytical server (listener) command until it receives a successful result. If the Campaign application continues to receive a non-success response after the maximum number of tries, a `Server Busy` error is displayed on the user interface.

Modify this parameter based on the Campaign Analytics server response time, the network speed and latency.

Default value

5

Campaign | logging

This category specifies the location of the Campaign `log4jConfig` properties file.

log4jConfig

Description

The Campaign web application uses the Apache log4j utility for logging configuration, debugging, and error information.

The `log4jConfig` property specifies the location of the Campaign log properties file, `campaign_log4j.properties`. Specify the path relative to the Campaign home directory, including the file name. Use forward slashes (/) for UNIX and backslashes (\) for Windows.

Default value

`./conf/campaign_log4j.properties`

Campaign | proxy

The Campaign, Engage, and UBX integration is supported with outbound proxy connections.

To access these properties, choose **Settings** > **Configuration**.

Proxy host name

Description

Specify the hostname or the IP address of your proxy server.

Proxy port number

Description

Specify the port number of your proxy server.

Proxy type

Description

Select the proxy server type.

Default value

HTTP

Valid value

HTTP, SOCK5

Data source for credentials

Description

Specify the datasource name that contains the proxy server user name and password details.

Platform user with data source for proxy credentials

Description

Specify the name of the Marketing Platform user that has the specified datasource in the **Data source for credentials** property.

Note: When you deploy Campaign on a WebLogic server and HTTP proxy is configured, you need to add the variable `DUseSunHttpHandler=true` in `JAVA_OPTION` to the `setDomainEnv.cmd` file.

eMessage configuration properties

This section describes the eMessage configuration properties found on the Configuration page.

eMessage | serverComponentsAndLocations | hostedServices

Define properties to specify the URLs for connecting to IBM Marketing Software Hosted Services. eMessage uses separate connections for uploading recipient lists, metadata that describes recipient lists, and for general communication sent to the hosted environment.

You must change the default values if you are connecting to IBM Marketing Software Hosted Services through the data center that is established by IBM in the United Kingdom. Consult IBM to determine the data center to which you are connected.

uiHostName

Description

The address that eMessage uses for all communication to IBM Marketing Software Hosted Services, except uploading recipient lists and related metadata.

Default value

`em.unicaondemand.com`

If you are connecting to the UK data center, change this value to `em-eu.unicaondemand.com`.

dataHostName

Description

The address that eMessage uses for uploading metadata that is related to recipient lists to IBM Marketing Software Hosted Services.

Default value

`em.unicaondemand.com`

If you are connecting to the UK data center, change this value to `em-eu.unicaondemand.com`.

ftpHostName

Description

The address that eMessage uses for uploading recipient list data (except list metadata) to IBM Marketing Software Hosted Services.

Default value

`ftp-em.unicaondemand.com`

If you are connecting to the UK data center, change this value to `ftp-em-eu.unicaondemand.com`.

eMessage | partitions | partition[n] | hostedAccountInfo

Define properties in this category to define user credentials for the database that contains account information that is required to access IBM Marketing Software Hosted Services. Values that you specify here must be defined as user settings in the Marketing Platform.

amUserForAcctCredentials

Description

Use this property to specify the Marketing Platform user that contains a Marketing Platform data source that specifies the account access credentials that are required to access IBM Marketing Software Hosted Services.

Default value

asm_admin

Valid Values

Any Marketing Platform user.

amDataSourceForAcctCredentials

Description

Use this property to specify the Marketing Platform data source that defines login credentials for IBM Marketing Software Hosted Services.

Default value

UNICA_HOSTED_SERVICES

Valid Values

A data source that is associated with the user you specify in `amUserForAcctCredentials`

eMessage | partitions | partition[n] | dataSources | systemTables

This category contains configuration properties that define the schema, connection settings, and login credentials for the database that contains the eMessage system tables in your network environment.

type

Description

Type of database that hosts the eMessage system tables.

Default value

No default value defined. You must define this property.

Valid Values

- SQLSERVER
- ORACLE9
- ORACLE10 (also used to indicate Oracle 11 databases)
- DB2

schemaName

Description

Name of the database schema for the eMessage system tables. This is the same as the schema name for the Campaign system tables.

You must include this schema name when referencing system tables in scripts.

Default value

dbo

jdbcBatchSize

Description

The number of execution requests JDBC runs on the database at a time.

Default value

10

Valid Values

An integer greater than 0.

jdbcClassName

Description

JDBC driver for system tables as defined in your Campaign web server.

Default value

No default value defined. You must define this property.

jdbcURI

Description

JDBC connection URI for system tables as defined in your Campaign web server.

Default value

No default value defined. You must define this property.

asmUserForDBCredentials

Description

Use this property to specify an IBM Marketing Software user that will be allowed to access the eMessage system tables.

Default value

No default value defined. You must define this property.

Valid Values

Any user defined in the Marketing Platform. This should typically be the name of the system user for Campaign

amDataSourceForDBCredentials

Description

Use this property to specify the data source that defines login credentials for the database that contains the eMessage system tables. This can be the same as the data source for the Campaign system tables.

Default value

UA_SYSTEM_TABLES

Valid Values

A Marketing Platform data source associated with the IBM Marketing Software user you specify in `asmUserForDBCredentials`

The data source specifies a database user and credentials used to access the eMessage system tables. If the default schema for the database user is not the schema that contains the system tables you must specify the system table schema in the JDBC connection used to access the system tables.

poolAcquireIncrement

Description

When the database connection pool runs out of connections, the number of new connections eMessage creates for the system tables. eMessage creates new connections up to the number specified in `poolMaxSize`.

Default value

1

Valid Values

An integer greater than 0.

poolIdleTestPeriod

Description

The number of seconds eMessage waits between testing idle connections to the eMessage system tables for activity.

Default value

100

Valid Values

An integer greater than 0.

poolMaxSize

Description

The maximum number of connections eMessage makes to the system tables. A value of zero (0) indicates there is no maximum.

Default value

100

Valid Values

An integer greater than or equal to 0.

poolMinSize

Description

The minimum number of connections eMessage makes to the system tables.

Default value

10

Valid Values

An integer greater than or equal to 0.

poolMaxStatements

Description

The maximum number of statements that eMessage stores in the PrepareStatement cache per connection to the system tables. Setting poolMaxStatements to zero (0) disables statement caching.

Default value

0

Valid Values

An integer equal to or greater than 0.

timeout

Description

The number of seconds eMessage maintains an idle database connection before dropping the connection.

If `poolIdleTestPeriod` is greater than 0, eMessage tests all idle, pooled, but unchecked-out connections, every `timeout` number of seconds.

If `poolIdleTestPeriod` is greater than `timeout`, the idle connections are dropped.

Default value

100

Valid Values

An integer equal to or greater than 0.

eMessage | partitions | partition[n] | recipientListUploader

This configuration category contains an optional property for the location of a user-defined script that runs in response to the actions or status of the Recipient List Uploader.

pathToTriggerScript

Description

You can create a script that triggers an action in response to the upload of a recipient list to IBM Marketing Software Hosted Services. For example, you can create a script to send an email alert to the list designer when the list upload completes successfully.

If you define a value for this property, eMessage passes status information about the Recipient List Uploader to the specified location. eMessagetakes no action if you leave this property blank.

Default value

No default value defined.

Valid Values

Any valid network path.

eMessage | partitions | partition[n] | responseContactTracker

Properties in this category specify behavior for the Response and Contact Tracker (RCT). The RCT retrieves and processes data for email contacts, email delivery, and recipient responses, such as link clicks and opens.

pauseCustomerPremisesTracking

Description

eMessage stores contact and response data in a queue in IBM Marketing Software Hosted Services. This property allows you to instruct the RCT to temporarily stop retrieving data from IBM Marketing Software Hosted Services. When you resume tracking, the RCT downloads the accumulated data.

Default value

False

Valid Values

True | False

waitTimeToCheckForDataAvailability

Description

The RCT periodically checks for new data regarding email contacts or recipient responses. This property allows you to specify how often, in seconds, the RCT checks for new data in IBM Marketing Software Hosted Services. The default value is 300 seconds, or every 5 minutes.

Default value

300

Valid Values

Any integer greater than 1.

perfLogInterval

Description

This property allows you to specify how often the RCT logs performance statistics to a log file. The value you enter determines the number of batches between log entries.

Default value

10

Valid Values

An integer greater than 0.

enableSeparatePartialResponseDataTracking

Description

This property determines if eMessage forwards partial email response data to the tracking tables in your local eMessage installation.

eMessage requires the Mailing Instance ID and Message Sequence Number to properly attribute email responses. When you enable separate partial response data tracking, eMessage places the incomplete responses in separate local tracking tables where you can review them or perform additional processing.

Default value

True

Valid Values

True | False

enableExecutionHistoryDataTracking

Description

This property controls whether you can download additional mailing execution history data from IBM Marketing Software Hosted Services.

By default, this property is set to **False**, to prevent download of additional data. When you set this property to **True**, you can download data about mailing runs that is not ordinarily entered to the eMessage system tables. You can use this supplementary information to help automate mailing and database management.

This property is hidden by default. You can display this configuration property in your local eMessage installation by running the `switch_config_visibility.bat` script, located in the `emessage\tools` directory.

Access to mailing execution history data is available by request from IBM. To request access to additional mailing execution history data, contact your IBM representative at `eacctsvc@us.ibm.com`.

Default value

False

Valid Values

True | False

Interact configuration properties

This section describes the Interact configuration properties found on the Configuration page.

Interact runtime environment configuration properties

This section describes all the configuration properties for the Interact runtime environment.

Interact | general

These configuration properties define general settings for your runtime environment environment, including the default logging level and the locale setting.

log4jConfig

Description

The location of the file containing the log4j properties. This path must be relative to the `INTERACT_HOME` environment variable. `INTERACT_HOME` is the location of the Interact installation directory.

Default value

`./conf/interact_log4j.properties`

asmUserForDefaultLocale

Description

The `asmUserForDefaultLocale` property defines the IBM Marketing Software user from which Interact derives its locale settings.

The locale settings define what language displays in the design time and what language advisory messages from the Interact API are in. If the locale setting does not match your machines operating system settings, Interact still functions, however the design time display and advisory messages may be in a different language.

Default value

`asm_admin`

Interact | general | learningTablesDataSource

These configuration properties define the data source settings for the built-in learning tables. You must define this data source if you are using Interact built-in learning.

If you create your own learning implementation using the Learning API, you can configure your custom learning implementation to read these values using the `ILearningConfig` interface.

jndiName

Description

Use this jndiName property to identify the Java Naming and Directory Interface (JNDI) data source that is defined in the application server (Websphere or WebLogic) for the learning tables accessed by Interact runtime servers.

The learning tables are created by the aci_lrntab ddl file and contain the following tables (among others): UACI_AttributeValue and UACI_OfferStats.

Default value

No default value defined.

type

Description

The database type for the data source used by the learning tables accessed by the Interact runtime servers.

The learning tables are created by the aci_lrntab ddl file and contain the following tables (among others): UACI_AttributeValue and UACI_OfferStats.

Default value

SQLServer

Valid Values

SQLServer | DB2 | ORACLE

connectionRetryPeriod

Description

The ConnectionRetryPeriod property specifies the amount of time in seconds Interact automatically retries the database connection request on failure for the learning tables. Interact automatically tries to reconnect to the database for this length of time before reporting a database error or failure. If the value is set to 0, Interact will retry indefinitely; if the value is set to -1, no retry will be attempted.

The learning tables are created by the aci_lrntab ddl file and contain the following tables (among others): UACI_AttributeValue and UACI_OfferStats.

Default value

-1

connectionRetryDelay

Description

The ConnectionRetryDelay property specifies the amount of time in seconds Interact waits before it tries to reconnect to the database after a failure for the learning tables. If the value is set to -1, no retry will be attempted.

The learning tables are created by the aci_lrntab ddl file and contain the following tables (among others): UACI_AttributeValue and UACI_OfferStats.

Default value

-1

schema

Description

The name of the schema containing the tables for the built-in learning module. Interact inserts the value of this property before all table names, for example, `UACI_IntChannel` becomes `schema.UACI_IntChannel`.

You do not have to define a schema. If you do not define a schema, Interact assumes that the owner of the tables is the same as the schema. You should set this value to remove ambiguity.

Default value

No default value defined.

Interact | general | prodUserDataSource

These configuration properties define the data source settings for the production profile tables. You must define this data source. This is the data source the runtime environment references when running interactive flowcharts after deployment.

jndiName

Description

Use this `jndiName` property to identify the Java Naming and Directory Interface (JNDI) data source that is defined in the application server (Websphere or WebLogic) for the customer tables accessed by Interact runtime servers.

Default value

No default value defined.

type

Description

The database type for the customer tables accessed by Interact runtime servers.

Default value

SQLServer

Valid Values

SQLServer | DB2 | ORACLE

aliasPrefix

Description

The `AliasPrefix` property specifies the way Interact forms the alias name that Interact creates automatically when using a dimension table and writing to a new table in the customer tables accessed by Interact runtime servers..

Note that each database has a maximum identifier length; check the documentation for the database you are using to be sure that the value you set does not exceed the maximum identifier length for your database.

Default value

A

connectionRetryPeriod

Description

The ConnectionRetryPeriod property specifies the amount of time in seconds Interact automatically retries the database connection request on failure for the runtime customer tables. Interact automatically tries to reconnect to the database for this length of time before reporting a database error or failure. If the value is set to 0, Interact will retry indefinitely; if the value is set to -1, no retry will be attempted.

Default value

-1

connectionRetryDelay

Description

The ConnectionRetryDelay property specifies the amount of time in seconds Interact waits before it tries to reconnect to the database after a failure for the Interact runtime customer tables. If the value is set to -1, no retry will be attempted.

Default value

-1

schema

Description

The name of the schema containing your profile data tables. Interact inserts the value of this property before all table names, for example, UACI_IntChannel becomes schema.UACI_IntChannel.

You do not have to define a schema. If you do not define a schema, Interact assumes that the owner of the tables is the same as the schema. You should set this value to remove ambiguity.

When you use a DB2 database, the schema name must be upper case.

Default value

No default value defined.

Interact | general | systemTablesDataSource

These configuration properties define the data source settings for the system tables for runtime environment. You must define this data source.

jndiName

Description

Use this jndiName property to identify the Java Naming and Directory Interface (JNDI) data source that is defined in the application server (Websphere or WebLogic) for the runtime environment tables.

The runtime environment database is the database populated with the aci_runtime and aci_populate_runtime dll scripts and, for example, contains the following tables (among others): UACI_CHOfferAttrib and UACI_DefaultedStat.

Default value

No default value defined.

type

Description

The database type for the runtime environment system tables.

The runtime environment database is the database populated with the `aci_runtime` and `aci_populate_runtime` dll scripts and, for example, contains the following tables (among others): `UACI_CHOfferAttrib` and `UACI_DefaultedStat`.

Default value

SQLServer

Valid Values

SQLServer | DB2 | ORACLE

connectionRetryPeriod

Description

The `ConnectionRetryPeriod` property specifies the amount of time in seconds Interact automatically retries the database connection request on failure for the runtime system tables. Interact automatically tries to reconnect to the database for this length of time before reporting a database error or failure. If the value is set to 0, Interact will retry indefinitely; if the value is set to -1, no retry will be attempted.

The runtime environment database is the database populated with the `aci_runtime` and `aci_populate_runtime` dll scripts and, for example, contains the following tables (among others): `UACI_CHOfferAttrib` and `UACI_DefaultedStat`.

Default value

-1

connectionRetryDelay

Description

The `ConnectionRetryDelay` property specifies the amount of time in seconds Interact waits before it tries to reconnect to the database after a failure for the Interact runtime system tables. If the value is set to -1, no retry will be attempted.

The runtime environment database is the database populated with the `aci_runtime` and `aci_populate_runtime` dll scripts and, for example, contains the following tables (among others): `UACI_CHOfferAttrib` and `UACI_DefaultedStat`.

Default value

-1

schema

Description

The name of the schema containing the tables for the runtime environment. Interact inserts the value of this property before all table names, for example, `UACI_IntChannel` becomes `schema.UACI_IntChannel`.

You do not have to define a schema. If you do not define a schema, Interact assumes that the owner of the tables is the same as the schema. You should set this value to remove ambiguity.

Default value

No default value defined.

Interact | general | systemTablesDataSource | loaderProperties

These configuration properties define the settings a database loader utility for the system tables for runtime environment. You need to define these properties if you are using a database loader utility only.

databaseName

Description

The name of the database the database loader connects to.

Default value

No default value defined.

LoaderCommandForAppend

Description

The `LoaderCommandForAppend` parameter specifies the command issued to invoke your database load utility for appending records to the contact and response history staging database tables in Interact. You need to set this parameter to enable the database loader utility for contact and response history data.

This parameter is specified as a full path name either to the database load utility executable or to a script that launches the database load utility. Using a script allows you to perform additional setup before invoking the load utility.

Most database load utilities require several arguments to be successfully launched. These can include specifying the data file and control file to load from and the database and table to load into. The tokens are replaced by the specified elements when the command is run.

Consult your database load utility documentation for the correct syntax to use when invoking your database load utility.

This parameter is undefined by default.

Tokens available to `LoaderCommandForAppend` are described in the following table.

Token	Description
<CONTROLFILE>	This token is replaced with the full path and filename to the temporary control file that Interact generates according to the template that is specified in the `LoaderControlFileTemplate` parameter.

Token	Description
<DATABASE>	This token is replaced with the name of the data source into which Interact is loading data. This is the same data source name used in the category name for this data source.
<DATAFILE>	This token is replaced with the full path and filename to the temporary data file created by Interact during the loading process. This file is in the Interact Temp directory, UNICA_ACTMPDIR.
<DBCOLUMNNUMBER>	This token is replaced with the column ordinal in the database.
<FIELDLENGTH>	This token is replaced with the length of the field being loaded into the database.
<FIELDNAME>	This token is replaced with the name of the field being loaded into the database.
<FIELDNUMBER>	This token is replaced with the number of the field being loaded into the database.
<FIELDTYPE>	This token is replaced with the literal "CHAR( )". The length of this field is specified between the (). If your database happens to not understand the field type, CHAR, you can manually specify the appropriate text for the field type and use the <FIELDLENGTH> token. For example, for SQLSVR and SQL2000 you would use "SQLCHAR(<FIELDLENGTH>)"
<NATIVETYPE>	This token is replaced with the type of database into which this field is loaded.
<NUMFIELDS>	This token is replaced with the number of fields in the table.
<PASSWORD>	This token is replaced with the database password from the current flowchart connection to the data source.
<TABLENAME>	This token is replaced with the database table name into which Interact is loading data.
<USER>	This token is replaced with the database user from the current flowchart connection to the data source.

Default value

No default value defined.

LoaderControlFileTemplateForAppend

Description

The LoaderControlFileTemplateForAppend property specifies the full path and filename to the control file template that has been previously configured in Interact. When this parameter is set, Interact dynamically builds a temporary control file based on the template that is specified here. The path and name of this temporary control file is available to the <CONTROLFILE> token that is available to the LoaderCommandForAppend property.

Before you use Interact in the database loader utility mode, you must configure the control file template that is specified by this parameter. The control file template supports the following tokens, which are dynamically replaced when the temporary control file is created by Interact.

See your database loader utility documentation for the correct syntax required for your control file. Tokens available to your control file template are the same as those for the LoaderControlFileTemplate property.

This parameter is undefined by default.

Default value

No default value defined.

LoaderDelimiterForAppend

Description

The LoaderDelimiterForAppend property specifies whether the temporary Interact data file is a fixed-width or delimited flat file, and, if it is delimited, the character or set of characters used as delimiters.

If the value is undefined, Interact creates the temporary data file as a fixed width flat file.

If you specify a value, it is used when the loader is invoked to populate a table that is not known to be empty. Interact creates the temporary data file as a delimited flat file, using the value of this property as the delimiter.

This property is undefined by default.

Default value

Valid Values

Characters, which you may enclose in double quotation marks, if desired.

LoaderDelimiterAtEndForAppend

Description

Some external load utilities require that the data file be delimited and that each line end with the delimiter. To accommodate this requirement, set the LoaderDelimiterAtEndForAppend value to TRUE, so that when the loader is invoked to populate a table that is not known to be empty, Interact uses delimiters at the end of each line.

Default value

FALSE

Valid Values

TRUE | FALSE

LoaderUseLocaleDP

Description

The LoaderUseLocaleDP property specifies, when Interact writes numeric values to files to be loaded by a database load utility, whether the locale-specific symbol is used for the decimal point.

Set this value to FALSE to specify that the period (.) is used as the decimal point.

Set this value to TRUE to specify that the decimal point symbol appropriate to the locale is used.

Default value

FALSE

Valid Values

TRUE | FALSE

Interact | general | testRunDataSource

These configuration properties define the data source settings for the test run tables for the Interact design environment. You must define this data source for at least one of your runtime environments. These are the tables used when you perform a test run of your interactive flowchart.

jndiName

Description

Use this jndiName property to identify the Java Naming and Directory Interface (JNDI) data source that is defined in the application server (Websphere or WebLogic) for the customer tables accessed by the design environment when executing interactive flowcharts test runs.

Default value

No default value defined.

type

Description

The database type for the customer tables accessed by the design environment when executing interactive flowcharts test runs.

Default value

SQLServer

Valid Values

SQLServer | DB2 | ORACLE

aliasPrefix

Description

The AliasPrefix property specifies the way Interact forms the alias name that Interact creates automatically when using a dimension table and

writing to a new table for the customer tables accessed by the design environment when executing interactive flowcharts test runs.

Note that each database has a maximum identifier length; check the documentation for the database you are using to be sure that the value you set does not exceed the maximum identifier length for your database.

Default value

A

connectionRetryPeriod

Description

The `ConnectionRetryPeriod` property specifies the amount of time in seconds Interact automatically retries the database connection request on failure for the test run tables. Interact automatically tries to reconnect to the database for this length of time before reporting a database error or failure. If the value is set to 0, Interact will retry indefinitely; if the value is set to -1, no retry will be attempted.

Default value

-1

connectionRetryDelay

Description

The `ConnectionRetryDelay` property specifies the amount of time in seconds Interact waits before it tries to reconnect to the database after a failure for the test run tables. If the value is set to -1, no retry will be attempted.

Default value

-1

schema

Description

The name of the schema containing the tables for interactive flowchart test runs. Interact inserts the value of this property before all table names, for example, `UACI_IntChannel` becomes `schema.UACI_IntChannel`.

You do not have to define a schema. If you do not define a schema, Interact assumes that the owner of the tables is the same as the schema. You should set this value to remove ambiguity.

Default value

No default value defined.

Interact | general | contactAndResponseHistoryDataSource

These configuration properties define the connection settings for the contact and response history data source required for the Interact cross-session response tracking. These settings are not related to the contact and response history module.

jndiName

Description

Use this jndiName property to identify the Java Naming and Directory Interface (JNDI) data source that is defined in the application server (WebSphere or WebLogic) for the contact and response history data source required for the Interact cross-session response tracking.

Default value

type

Description

The database type for the data source used by the contact and response history data source required for the Interact cross-session response tracking.

Default value

SQLServer

Valid Values

SQLServer | DB2 | ORACLE

connectionRetryPeriod

Description

The ConnectionRetryPeriod property specifies the amount of time in seconds Interact automatically retries the database connection request on failure for the Interact cross-session response tracking. Interact automatically tries to reconnect to the database for this length of time before reporting a database error or failure. If the value is set to 0, Interact will retry indefinitely; if the value is set to -1, no retry will be attempted.

Default value

-1

connectionRetryDelay

Description

The ConnectionRetryDelay property specifies the amount of time in seconds Interact waits before it tries to reconnect to the database after a failure for the Interact cross-session response tracking. If the value is set to -1, no retry will be attempted.

Default value

-1

schema

Description

The name of the schema containing the tables for the Interact cross-session response tracking. Interact inserts the value of this property before all table names, for example, UACI_IntChannel becomes schema.UACI_IntChannel.

You do not have to define a schema. If you do not define a schema, Interact assumes that the owner of the tables is the same as the schema. You should set this value to remove ambiguity.

Default value

No default value defined.

Interact | general | idsByType

These configuration properties define settings for ID numbers used by the contact and response history module.

initialValue

Description

The initial ID value used when generating IDs using the UACI_IDsByType table.

Default value

1

Valid Values

Any value greater than 0.

retries

Description

The number of retries before generating an exception when generating IDs using the UACI_IDsByType table.

Default value

20

Valid Values

Any integer greater than 0.

Interact | flowchart

This section defines configuration settings for interactive flowcharts.

defaultDateFormat

Description

The default date format used by Interact to convert Date to String and String to Date.

Default value

MM/dd/yy

idleFlowchartThreadTimeoutInMinutes

Description

The number of minutes Interact allows a thread dedicated to an interactive flowchart to be idle before releasing the thread.

Default value

5

idleProcessBoxThreadTimeoutInMinutes

Description

The number of minutes Interact allows a thread dedicated to an interactive flowchart process to be idle before releasing the thread.

Default value

5

maxSizeOfFlowchartEngineInboundQueue

Description

The maximum number of flowchart run requests Interact holds in queue. If this number of requests is reached, Interact will stop taking requests.

Default value

1000

maxNumberOfFlowchartThreads

Description

The maximum number of threads dedicated to interactive flowchart requests.

Default value

25

maxNumberOfProcessBoxThreads

Description

The maximum number of threads dedicated to interactive flowchart processes.

Default value

50

maxNumberOfProcessBoxThreadsPerFlowchart

Description

The maximum number of threads dedicated to interactive flowchart processes per flowchart instance.

Default value

3

minNumberOfFlowchartThreads

Description

The minimum number of threads dedicated to interactive flowchart requests.

Default value

10

minNumberOfProcessBoxThreads

Description

The minimum number of threads dedicated to interactive flowchart processes.

Default value

20

sessionVarPrefix

Description

The prefix for session variables.

Default value

SessionVar

Interact | flowchart | ExternalCallouts

This section defines the class settings for custom external callouts you have written with the external callout API.

class

Description

The name of the Java class represented by this external callout.

This is the Java class that you can access with the IBM Macro `EXTERNALCALLOUT`.

Default value

No default value defined.

classpath

Description

The classpath for the Java class represented by this external callout. The classpath must reference jar files on the runtime environment server. If you are using a server group and all runtime servers are using the same Marketing Platform, every server must have a copy of the jar file in the same location. The classpath must consist of absolute locations of jar files, separated by the path delimiter of the operating system of the runtime environment server, for example a semi-colon (;) on Windows and a colon (:) on UNIX systems. Directories containing class files are not accepted. For example, on a Unix system: `/path1/file1.jar:/path2/file2.jar`.

This classpath must be less than 1024 characters. You can use the manifest file in a .jar file to specify other .jar files so only one .jar file has to appear in your class path

This is the Java class that you can access with the IBM Macro `EXTERNALCALLOUT`.

Default value

No default value defined.

value

Description

The value for any parameter required by the class for the external callout.

Default value

No default value defined.

Example

If the external callout requires host name of an external server, create a parameter category named `host` and define the `value` property as the server name.

Interact | monitoring

This set of configuration properties enables you to define JMX monitoring settings. You need to configure these properties only if you are using JMX monitoring. There are separate JMX monitoring properties to define for the contact and response history module in the configuration properties for Interact design environment.

protocol

Description

Define the protocol for the Interact messaging service.

If you choose JMXMP you must include the following JAR files in your class path in order:

`Interact/lib/InteractJMX.jar;Interact/lib/jmxremote_optional.jar`

Default value

JMXMP

Valid Values

JMXMP | RMI

port

Description

The port number for the messaging service.

Default value

9998

enableSecurity

Description

A boolean which enables or disables JMXMP messaging service security for the Interact runtime server. If set to `true`, you must supply a user name and password to access the Interact runtime JMX service. This user credential is authenticated by the Marketing Platform for the runtime server. Jconsole does not allow empty password login.

This property has no effect if the protocol is RMI. This property has no effect on JMX for Campaign (the Interact design time).

Default value

True

Valid Values

True | False

Interact | monitoring | activitySubscribers

This set of configuration properties enables the root node for the settings that are related to remote subscribers that can receive periodic update on basic performance data in the Interact runtime environment.

heartbeatPeriodInSecs

Description

The interval in seconds when each runtime instance sends an update to subscribers.

Default value

60

Interact | monitoring | activitySubscribers | (target)

(target)

Description

The root node for the settings of a subscriber.

URL

Description

The URL of this subscriber. This endpoint must be able to accept JSON messages transported through HTTP.

continuousErrorsForAbort

Description

The number of continuous failed updates before the runtime instance stops sending more updates to this subscriber.

Default value

5

timeoutInMillis

Description

The time-out in milliseconds the send process times out during sending update to this subscriber.

Default value

1000

Valid Values

Enabled

Description

Whether this subscriber is enabled or disabled.

Default value

True

Valid Values

True or False

type

Description

The type of this data store. When this option is selected, the parameter **className** must be added with the value being the fully qualified name of this implementation class. **classPath** needs to be added with the URI of the JAR file if it is not in the class path of the Interact run time.

Default value

InteractLog

Valid Values

InteractLog, RelationalDB, and Custom

jmxInclusionCycles

Description

The interval in the multiplier of **heartbeatPeriodInSecs** that detailed JMX statistics are sent to this subscriber.

Default value

5

Valid Values

Interact | profile

This set of configuration properties control several of the optional offer serving features, including offer suppression and score override.

enableScoreOverrideLookup

Description

If set to True, Interact loads the score override data from the scoreOverrideTable when creating a session. If False, Interact does not load the marketing score override data when creating a session.

If true, you must also configure the Interact | profile | Audience Levels | (Audience Level) | scoreOverrideTable property. You need to define the scoreOverrideTable property for the audience levels you require only. Leaving the scoreOverrideTable blank for an audience level disables the score override table for the audience level.

Default value

False

Valid Values

True | False

enableOfferSuppressionLookup

Description

If set to True, Interact loads the offer suppression data from the offerSuppressionTable when creating a session. If False, Interact does not load the offer suppression data when creating a session.

If true, you must also configure the Interact | profile | Audience Levels | (Audience Level) | offerSuppressionTable property. You need to define the enableOfferSuppressionLookup property for the audience levels you require only.

Default value

False

Valid Values

True | False

enableProfileLookup

Description

In a new installation of Interact, this property is deprecated. In an upgraded installation of Interact, this property is valid until the first deployment.

The load behavior for a table used in an interactive flowchart but not mapped in the interactive channel. If set to `True`, Interact loads the profile data from the `profileTable` when creating a session.

If true, you must also configure the `Interact | profile | Audience Levels | (Audience Level) | profileTable` property.

The **Load this data in to memory when a visit session starts** setting in the interactive channel table mapping wizard overrides this configuration property.

Default value

`False`

Valid Values

True | False

defaultOfferUpdatePollPeriod

Description

The number of seconds the system waits before updating the default offers in the cache from the default offers table. If set to -1, the system doesn't update the default offers in the cache after the initial list is loaded into the cache when the runtime server starts.

Default value

-1

Interact | profile | Audience Levels | [AudienceLevelName]

This set of configuration properties enables you to define the table names required for additional Interact features. You are only required to define the table name if you are using the associated feature.

New category name

Description

The name of your audience level.

scoreOverrideTable

Description

The name of the table containing the score override information for this audience level. This property is applicable if you have set `enableScoreOverrideLookup` to true. You have to define this property for the audience levels for which you want to enable a score override table. If you have no score override table for this audience level, you can leave this property undefined, even if `enableScoreOverrideLookup` is set to true.

Interact looks for this table in the customer tables accessed by Interact runtime servers, defined by the `prodUserDataSource` properties.

If you have defined the schema property for this data source, Interact prepends this table name with the schema, for example, schema.UACI_ScoreOverride. If you enter a fully-qualified name, for example, mySchema.UACI_ScoreOverride, Interact does not prepend the schema name.

Default value

UACI_ScoreOverride

offerSuppressionTable

Description

The name of the table containing the offer suppression information for this audience level. You have to define this property for the audience levels for which you want to enable an offer suppression table. If you have no offer suppression table for this audience level, you can leave this property undefined. If enableOfferSuppressionLookup is set to true, this property must be set to a valid table.

Interact looks for this table in the customer tables accessed by runtime servers, defined by the prodUserDataSource properties.

Default value

UACI_BlackList

contactHistoryTable

Description

The name of the staging table for the contact history data for this audience level.

This table is stored in the runtime environment tables (systemTablesDataSource).

If you have defined the schema property for this data source, Interact prepends this table name with the schema, for example, schema.UACI_CHStaging. If you enter a fully-qualified name, for example, mySchema.UACI_CHStaging, Interact does not prepend the schema name.

If contact history logging is disabled, this property does not need to be set.

Default value

UACI_CHStaging

chOfferAttribTable

Description

The name of the contact history offer attributes table for this audience level.

This table is stored in the runtime environment tables (systemTablesDataSource).

If you have defined the schema property for this data source, Interact prepends this table name with the schema, for example, schema.UACI_CHOfferAttrib. If you enter a fully-qualified name, for example, mySchema.UACI_CHOfferAttrib, Interact does not prepend the schema name.

If contact history logging is disabled, this property does not need to be set.

Default value

UACI_CHOfferAttrib

responseHistoryTable

Description

The name of the response history staging table for this audience level.

This table is stored in the runtime environment tables (`systemTablesDataSource`).

If you have defined the `schema` property for this data source, Interact prepends this table name with the schema, for example, `schema.UACI_RHStaging`. If you enter a fully-qualified name, for example, `mySchema.UACI_RHStaging`, Interact does not prepend the schema name.

If response history logging is disabled, this property does not need to be set.

Default value

UACI_RHStaging

crossSessionResponseTable

Description

The name of the table for this audience level required for cross-session response tracking in the contact and response history tables accessible for the response tracking feature.

If you have defined the `schema` property for this data source, Interact prepends this table name with the schema, for example, `schema.UACI_XSessResponse`. If you enter a fully-qualified name, for example, `mySchema.UACI_XSessResponse`, Interact does not prepend the schema name.

If cross session response logging is disabled, this property does not need to be set.

Default value

`UACI_XSessResponse`

userEventLoggingTable

Description

This is the name of the database table that is used for logging user-defined event activities. Users defined events on the Events tab of the Interactive Channel summary pages in the Interact interface. The database table you specify here stores information such as the event ID, name, how many times this event occurred for this audience level since the last time the event activity cache was flushed, and so on.

If you have defined the `schema` property for this data source, Interact prepends this table name with the schema, for example, `schema.UACI_UserEventActivity`. If you enter a fully-qualified name, for example, `mySchema.UACI_UserEventActivity`, Interact does not prepend the schema name.

Default value

UACI_UserEventActivity

patternStateTable

Description

This is the name of the database table that is used for logging event pattern states, such as whether the pattern condition has been met or not, whether the pattern is expired or disabled, and so on.

If you have defined the schema property for this data source, Interact prepends this table name with the schema, for example, schema.UACI_EventPatternState. If you enter a fully-qualified name, for example, mySchema.UACI_EventPatternState, Interact does not prepend the schema name.

A patternStateTable is required for each audience level even if you do not use event patterns. The patternStateTable is based on the ddl of the included UACI_EventPatternState. The following is an example where the audience ID has two components; ComponentNum and ComponentStr.

```
CREATE TABLE UACI_EventPatternState_Composite
(
   UpdateTime bigint NOT NULL,
   State varbinary(4000),
   ComponentNum bigint NOT NULL,
   ComponentStr nvarchar(50) NOT NULL,
   CONSTRAINT PK_CustomerPatternState_Composite PRIMARY KEY
   (ComponentNum,ComponentStr,UpdateTime)
)
```

Default value

UACI_EventPatternState

Interact | profile | Audience Levels | [AudienceLevelName] | Offers by Raw SQL

This set of configuration properties enables you to define the table names required for additional Interact features. You are only required to define the table name if you are using the associated feature.

enableOffersByRawSQL

Description

If set to True, Interact enables the offersBySQL feature for this audience level that allows you to configure SQL code to be executed to create a desired set of candidate offers at runtime.. If False, Interact does not use the offersBySQL feature.

If you set this property to true, you may also configure the Interact | profile | Audience Levels | (Audience Level) | Offers by Raw SQL | SQL Template property to define one or more SQL templates.

Default value

False

Valid Values

True | False

cacheSize

Description

Size of cache used to store results of the OfferBySQL queries. Note that using a cache may have negative impact if query results are unique for most sessions.

Default value

-1 (off)

Valid Values

-1 | Value

cacheLifeInMinutes

Description

If the cache is enabled, this indicates the number of minutes before the system will clear the cache to avoid staleness.

Default value

-1 (off)

Valid Values

-1 | Value

defaultSQLTemplate

Description

The name of the SQL template to use if one is not specified via the API calls.

Default value

None

Valid Values

SQL template name

name

Configuration category

Interact | profile | Audience Levels | [AudienceLevelName] | Offers by Raw SQL | (SQL Templates)

Description

The name you want to assign to this SQL query template. Enter a descriptive name that will be meaningful when you use this SQL template in API calls. Note that if you use a name here that is *identical* to a name defined in the Interact List process box for an offerBySQL treatment, the SQL in the process box will be used rather than the SQL you enter here.

Default value

None

SQL

Configuration category

`Interact | profile | Audience Levels | [AudienceLevelName] | Offers by Raw SQL | (SQL Templates)`

Description

Contains the SQL query to be called by this template. The SQL query may contain references to variable names that are part of the visitor's session data (profile). For example, `select * from MyOffers where category = ${preferredCategory}` would rely on the session containing a variable named `preferredCategory`.

You should configure the SQL to query the specific offer tables you created during design time for use by this feature. Note that stored procedures are not supported here.

Default value

None

Interact | profile | Audience Levels | [AudienceLevelName | Profile Data Services | [DataSource]

This set of configuration properties enables you to define the table names required for additional Interact features. You are only required to define the table name if you are using the associated feature. The Profile Data Services category provides information about a built-in data source (called Database) that is created for all audience levels, and which is pre-configured with a priority of 100. However, you can choose to modify or disable it. This category also contains a template for additional external data sources. When you click the template called **External Data Services** you can complete the configuration settings described here.

New category name

Description

(Not available for the default Database entry.) The name of the data source you are defining. The name you enter here must be unique among the data sources for the same audience level.

Default value

None

Valid Values

Any text string is allowed.

enabled

Description

If set to `True`, this data source is enabled for the audience level to which it is assigned. If `False`, Interact does not use this data source for this audience level.

Default value

`True`

Valid Values

True | False

className

Description

(Not available for the default Database entry.) The fully-qualified name of the data source class that implements `IInteractProfileDataService`.

Default value

None.

Valid Values

A string providing a fully-qualified class name.

classPath

Description

(Not available for the default Database entry.) An optional configuration setting providing the path to load this data source implementation class. If you omit it, the class path of the containing application server is used by default.

Default value

Not shown, but the class path of the containing application server is used by default if no value is provided here.

Valid Values

A string providing the class path.

priority

Description

The priority of this data source within this audience level. It has to be a unique value among all of the data sources for each audience level. (That is, if a priority is set to 100 for a data source, no other data source within the audience level may have a priority of 100.)

Default value

`100` for the default Database, `200` for user-defined data source

Valid Values

Any non-negative integer is allowed.

Interact | offerserving

These configuration properties define the generic learning configuration properties. If you are using built-in learning, to tune your learning implementation, use the configuration properties for the design environment.

offerTieBreakMethod

Description

The `offerTieBreakMethod` property defines the behavior of offer serving when two offers have equivalent (tied) scores. If you set this property to its default value of Random, Interact presents a random choice from among the offers that have equivalent scores. If you set this configuration to Newer Offer, Interact serves up the newer offer (based on having a higher offer ID) ahead of the older offer (lower offer ID) in the case where the scores among the offers are the same.

Note:

Interact has an optional feature that allows the administrator to configure the system to return the offers in random order independent of the score, by setting the `percentRandomSelection` option (`Campaign | partitions | [partition_number] | Interact | learning | percentRandomSelection`). The `offerTieBreakMethod` property described here is used only when `percentRandomSelection` is set to zero (disabled).

Default value

Random

Valid Values

Random | Newer Offer

optimizationType

Description

The `optimizationType` property defines whether Interact uses a learning engine to assist with offer assignments. If set to `NoLearning`, Interact does not use learning. If set to `BuiltInLearning`, Interact uses the Bayesian learning engine built with Interact. If set to `ExternalLearning`, Interact uses a learning engine you provide. If you select `ExternalLearning`, you must define the `externalLearningClass` and `externalLearningClassPath` properties.

Default value

NoLearning

Valid Values

NoLearning | BuiltInLearning | ExternalLearning

segmentationMaxWaitTimeInMS

Description

The maximum number of milliseconds that the runtime server waits for an interactive flowchart to complete before getting offers.

Default value

`5000`

treatmentCodePrefix

Description

The prefix prepended to treatment codes.

Default value

No default value defined.

effectiveDateBehavior

Description

Determines whether Interact should use an offer's effective date in filtering out offers that are presented to a visitor. Values include:

- -1 tells Interact to ignore the effective date on the offer.

0 tells Interact to use the effective date to filter the offer, so that if the offer effective date is earlier than or equal to the current date, the offer effective date, the offer is served to visitors.

If there is an **effectiveDateGracePeriod** value set, the grace period is also applied to determine whether to serve the offer.

- Any positive integer tells Interact to use the current date plus the value of this property to determine whether to serve the offer to visitors, so that if the offer effective date is earlier than the current date plus the value of this property, the offer is served to visitors.

  If there is an **effectiveDateGracePeriod** value set, the grace period is also applied to determine whether to serve the offer.

Default value

-1

effectiveDateGracePeriodOfferAttr

Description

Specifies the name of the custom attribute in an offer definition that indicates the effective date grace period. For example, you might configure this property with a value of `AltGracePeriod`. You would then define offers with a custom attribute called `AltGracePeriod` that is used to specify the number of days to use as a grace period with the **effectiveDateBehavior** property.

Suppose you create a new offer template with an effective date of 10 days from the current date, and include a custom attribute called `AltGracePeriod`. When you create an offer using the template, if you set the value of `AltGracePeriod` to 14 days, the offer would be served to visitors, because the current date is within the grace period of the offer effective date.

Default value

Blank

alwaysLogLearningAttributes

Description

Indicates whether Interact should write information about visitor attributes used by the learning module to the log files. Note that settings this value to true may affect learning performance and log file sizes.

Default value

False

Interact | offerserving | Built-in Learning Config

These configuration properties define the database write settings for built-in learning. To tune your learning implementation, use the configuration properties for the design environment.

version

Description

You can select 1 or 2. Version 1 is the basic configuration version that does not use parameters to set thread and record limits. Version 2 is the enhanced configuration version that lets you set thread and record

parameter to improve performance. These parameters perform aggregation and deletion when these parameter limits are reached.

Default value

1

insertRawStatsIntervalInMinutes

Description

The number of minutes the Interact learning module waits before inserting more rows into the learning staging tables. You may need to modify this time based on the amount of data the learning module is processing in your environment.

Default value

5

Valid Values

A positive integer

aggregateStatsIntervalInMinutes

Description

The number of minutes the Interact learning module waits between aggregating data in the learning stats tables. You may need to modify this time based on the amount of data the learning module is processing in your environment.

Default value

15

Valid Values

An integer greater than zero.

autoAdjustPercentage

Description

The value that determines the percentage of data the run of aggregation tries to process based on the metrics of the previous run. By default, this value is set to zero, which means the aggregator processes all staging records, and this auto adjustment functionality is disabled.

Default value

0

Valid Values

A number between 0 and 100.

enableObservationModeOnly

Description

If set to `True`, enables a learning mode where Interact collects data for learning without using that data for recommendations or offer arbitration. This allows you to operate self-learning in a startup mode until you determine that enough data is collected for recommendations.

Default value

False

Valid Values

True | False

excludeAbnormalAttribute

Description

The setting that determines whether to mark those attributes as invalid. If set to `IncludeAttribute`, abnormal attributes are included not marked as invalid. If set to `ExcludeAttribute`, abnormal attributes are excluded and marked as invalid.

Default value

IncludeAttribute

Valid Values

IncludeAttribute | ExcludeAttribute

saveOriginalValues

Description

You can set the values as “All Values”, “Binned Values”, or “None”. This will control what values will be logged in table UACI_LearningAttributeHist.

If “All Values” is selected then all learning attributes will be logged in the table. If this parameter is set to “Binned Values” then only those attributes will be logged in the table for which bins are created under “Interact-> Global Learning”.

If set to “None” no values will be logged in UACI_LearningAttributeHist .

By default this is set to “None”.

Default value

None

Valid Values

All Values | Binned Values | None

Interact | offerserving | Built-in Learning Config | Parameter Data | [parameterName]

These configuration properties define any parameters for your external learning module.

numberOfThreads

Description

The maximum number of threads the learning aggregator uses to process the data. A valid value is a positive integer, and should not be more than the maximum number of connections that are configured in the learning data source. This parameter is used only by aggregator version 2.

Default value

10

maxLogTimeSpanInMin

Description

If aggregator version 1 is selected, you can process the staging records in iterations to avoid overly large database batches. In this case, those staging records are processed by chunks; iteration by iteration in a single aggregation cycle. The value of this parameter specifies the maximum time span of staging records the aggregator tries to process in each iteration. This time span is based on LogTime field that is associated to each staging record, and only the records whose LogTime falls into the earliest time window is processed. A valid value is an integer that is not negative. If the value is 0, there is no limit, which means all the staging records are processed in a single iteration.

Default value

0

maxRecords

Description

If aggregator version 2 is selected, you can process the staging records in iterations to avoid overly large database batches. In this case, those staging records are processed in chunks; iteration by iteration in a single aggregation cycle. The value of this parameter specifies the maximum number of staging records the aggregator tries to process in each iteration. A valid value is an integer that is not negative. If the value is 0, there is no limit, which means all the staging records are processed in a single iteration.

Default value

0

value

Description

The value for any parameter that is required by the class for a built-in learning module.

Default value

No default value defined.

Interact | offerserving | External Learning Config

These configuration properties define the class settings for an external learning module you wrote using the learning API.

class

Description

If `optimizationType` is set to `ExternalLearning`, set `externalLearningClass` to the class name for the external learning engine.

Default value

No default value defined.

Availability

This property is applicable only if optimizationType is set to ExternalLearning.

classPath

Description

If optimizationType is set to ExternalLearning, set externalLearningClass to the classpath for the external learning engine.

The classpath must reference jar files on the runtime environment server. If you are using a server group and all runtime servers are using the same Marketing Platform, every server must have a copy of the jar file in the same location. The classpath must consist of absolute locations of jar files, separated by the path delimiter of the operating system of the runtime environment server, for example a semi-colon (;) on Windows and a colon (:) on UNIX systems. Directories containing class files are not accepted. For example, on a Unix system: /path1/file1.jar:/path2/file2.jar.

This classpath must be less than 1024 characters. You can use the manifest file in a .jar file to specify other .jar files so only one .jar file has to appear in your class path

Default value

No default value defined.

Availability

This property is applicable only if optimizationType is set to ExternalLearning.

Interact | offerserving | External Learning Config | Parameter Data | [parameterName]

These configuration properties define any parameters for your external learning module.

value

Description

The value for any parameter required by the class for an external learning module.

Default value

No default value defined.

Example

If the external learning module requires a path to an algorithm solver application, you would create a parameter category called solverPath and define the value property as the path to the application.

Interact | offerserving | Constraints

These configuration properties define the constraints placed upon the offer serving process.

maxOfferAllocationInMemoryPerInstance

Description

The size of a block of offers. Interact keeps a pool of offers in memory so that the system does not have to query to database each time an offer is

returned. Every time an offer is returned, the pool is adjusted. When the pool is exhausted, Interact gets another block of offers to fill the pool.

Default value

1000

Valid Values

An integer greater than 0.

maxDistributionPerIntervalPerInstanceFactor

Description

The constraint percentage for a given offer allocation for a runtime server to support the distribution across runtime servers.

Default value

100

Valid Values

An integer between 0 and 100.

constraintCleanupIntervalInDays

Description

How often the disabled counts from the UACI_OfferCount table are cleaned up. A value less than 1 disables this feature.

Default value

7

Valid Values

An integer greater than 0.

Interact | services

The configuration properties in this category define settings for all the services which manage collecting contact and response history data and statistics for reporting and writing to the runtime environment system tables.

externalLoaderStagingDirectory

Description

This property defines the location of the staging directory for a database load utility.

Default value

No default value defined.

Valid Values

A path relative to the Interact installation directory or an absolute path to a staging directory.

If you enable a database load utility, you must set the `cacheType` property in the `contactHist` and `responstHist` categories to `External Loader File`.

Interact | services | contactHist

The configuration properties in this category define the settings for the service that collects data for the contact history staging tables.

enableLog

Description

If `true`, enables the service which collects data for recording the contact history data. If `false`, no data is collected.

Default value

True

Valid Values

True | False

cacheType

Description

Defines whether the data collected for contact history is kept in memory (`Memory Cache`) or in a file (`External Loader file`). You can use `External Loader File` only if you have configured Interact to use a database loader utility.

If you select `Memory Cache`, use the `cache` category settings. If you select `External Loader File`, use the `fileCache` category settings.

Default value

Memory Cache

Valid Values

Memory Cache | External Loader File

Interact | services | contactHist | cache

The configuration properties in this category define the cache settings for the service that collects data for the contact history staging table.

threshold

Description

The number of records accumulated before the flushCacheToDB service writes the collected contact history data to the database.

Default value

100

insertPeriodInSecs

Description

The number of seconds between forced writes to the database.

Default value

3600

Interact | services | contactHist | fileCache

The configuration properties in this category define the cache settings for the service that collects contact history data if you are using a database loader utility.

threshold

Description

The number of records accumulated before the flushCacheToDB service writes the collected contact history data to the database.

Default value

100

insertPeriodInSecs

Description

The number of seconds between forced writes to the database.

Default value

3600

Interact | services | defaultedStats

The configuration properties in this category define the settings for the service that collects the statistics regarding the number of times the default string for the interaction point was used.

enableLog

Description

If `true`, enables the service that collects the statistics regarding the number of times the default string for the interaction point was used to the `UACI_DefaultedStat` table. If `false`, no default string statistics are collected.

If you are not using IBM reporting, you can set this property to `false` since the data collection is not required.

Default value

True

Valid Values

True | False

Interact | services | defaultedStats | cache

The configuration properties in this category define the cache settings for the service that collects the statistics regarding the number of times the default string for the interaction point was used.

threshold

Description

The number of records accumulated before the flushCacheToDB service writes the collected default string statistics to the database.

Default value

100

insertPeriodInSecs

Description

The number of seconds between forced writes to the database.

Default value

3600

Interact | services | eligOpsStats

The configuration properties in this category define the settings for the service that writes the statistics for eligible offers.

enableLog

Description

If `true`, enables the service that collects the statistics for eligible offers. If `false`, no eligible offer statistics are collected.

If you are not using IBM reporting, you can set this property to `false` since the data collection is not required.

Default value

True

Valid Values

True | False

Interact | services | eligOpsStats | cache

The configuration properties in this category define the cache settings for the service that collects the eligible offer statistics.

threshold

Description

The number of records accumulated before the flushCacheToDB service writes the collected eligible offer statistics to the database.

Default value

100

insertPeriodInSecs

Description

The number of seconds between forced writes to the database.

Default value

3600

Interact | services | eventActivity

The configuration properties in this category define the settings for the service that collects the event activity statistics.

enableLog

Description

If `true`, enables the service that collects the event activity statistics. If `false`, no event statistics are collected.

If you are not using IBM reporting, you can set this property to `false` since the data collection is not required.

Default value

True

Valid Values

True | False

Interact | services | eventActivity | cache

The configuration properties in this category define the cache settings for the service that collects the event activity statistics.

threshold

Description

The number of records accumulated before the flushCacheToDB service writes the collected event activity statistics to the database.

Default value

100

insertPeriodInSecs

Description

The number of seconds between forced writes to the database.

Default value

3600

Interact | services | eventPattern

The configuration properties in the `eventPattern` category define the settings for the service that collects the event pattern activity statistics.

persistUnknownUserStates

Description

Determines whether the event pattern states for an unknown audience ID (visitor) is retained in the database. By default, when a session ends, the statuses of all the updated event patterns associated with the visitor's audience ID are stored in the database, provided that the audience ID is known (that is, the visitor's profile can be found in the profile data source).

The `persistUnknownUserStates` property determines what happens if the audience ID is not known. By default, this property is set to `False`, and for unknown audience IDs, the event pattern states are discarded at the end of the session.

If you set this property to `True`, the event pattern states of unknown users (whose profile cannot be find in the configured profile data service) will be persisted.

Default value

False

Valid Values

True | False

mergeUnknowUserInSessionStates

Description

Determines how the event pattern states for unknown audience IDs (visitors) are retained. If the audience ID switches in the middle of a session, Interact tries to load the saved event pattern states for the new

audience ID from the database table. When the audience ID was unknown previously, and you set the `mergeUnknowUserInSessionStates` property is to `True`, the user event activities belonging to the previous audience ID in the same session will be merged into the new audience ID.

Default value

False

Valid Values

True | False

enableUserEventLog

Description

Determines whether user event activities are logged in the database.

Default value

False

Valid Values

True | False

Interact | services | eventPattern | userEventCache

The configuration properties in the `userEventCache` category define the settings that determine when event activity is moved from the cache to persist in the database.

threshold

Description

Determines the maximum number of event pattern states that can be stored in the event pattern state cache. When the limit is reached, the least-recently used states are flushed from the cache.

Default value

100

Valid Values

The desired number of event pattern states to retain in the cache.

insertPeriodInSecs

Description

Determines the maximum length of time in seconds that user event activities are queued in memory. When the time limit specified by this property is reached, those activities are persisted into the database.

Default value

3600 (60 minutes)

Valid Values

The desired number of seconds.

Interact | services | eventPattern | advancedPatterns

The configuration properties in this category control whether integration with Interact Advanced Patterns is enabled, and they define the timeout intervals for connections with Interact Advanced Patterns.

enableAdvancedPatterns

Description

If `true`, enables integration with Interact Advanced Patterns. If `false`, integration is not enabled. If integration was previously enabled, Interact uses the most recent pattern states received from Interact Advanced Patterns.

Default value

True

Valid Values

True | False

connectionTimeoutInMilliseconds

Description

Maximum time it can take to make an HTTP connection from the Interact real time environment to Interact Advanced Patterns. If the request times out, Interact uses the last saved data from patterns.

Default value

30

readTimeoutInMilliseconds

Description

After an HTTP connection is established between the Interact real time environment and Interact Advanced Patterns,and a request is sent to the Interact Advanced Patterns to get the status of an event pattern, the maximum time it can take to receive data. If the request times out, Interact uses the last saved data from patterns.

Default value

100

connectionPoolSize

Description

Size of the HTTP connection pool for communication between the Interact real time environment and Interact Advanced Patterns.

Default value

10

Interact | services | eventPattern | advancedPatterns | autoReconnect

The configuration properties in this category specify parameters for the automatic reconnection feature in the integration with Interact Advanced Patterns.

enable

Description

Determines whether the system to reconnects automatically if connection problems occur between the Interact real time environment and Interact Advanced Patterns. The default value of **True** enables this feature.

Default value

True

Valid Values

True | False

durationInMinutes

Description

This property specifies the time interval, in minutes, during which the system to evaluates repeated connection problems occurring between the Interact real time environment and Interact Advanced Patterns.

Default value

10

numberOfFailuresBeforeDisconnect

Description

This property specifies the number of connection failures allowed during the specified time period before the system automatically disconnects from Interact Advanced Patterns.

Default value

3

consecutiveFailuresBeforeDisconnect

Description

Determines whether the automatic reconnection feature evaluates only consecutive failures of the connection between the Interact real time environment with Interact Advanced Patterns. If you set this value to **False**, all failures within the specified time interval are evaluated.

Default value

True

sleepBeforeReconnectDurationInMinutes

Description

The system waits the number of minutes specified in this property before reconnecting after the system disconnects due to repeated failures as specified in the other properties in this category.

Default value

5

sendNotificationAfterDisconnect

Description

This property determines whether the system sends an email notification when a connection failure occurs. The notification message includes the Interact real time instance name for which failure occurred and the amount of time before reconnection occurs, as specified in the **sleepBeforeReconnectDurationInMinutes** property. The default value of **True** means that notifications are sent.

Default value

True

Interact | services | customLogger

The configuration properties in this category define the settings for the service that collects custom data to write to a table (an event which uses the `UACICustomLoggerTableName` event parameter).

enableLog

Description

If `true`, enables the custom log to table feature. If `false`, the `UACICustomLoggerTableName` event parameter has no effect.

Default value

True

Valid Values

True | False

Interact | services | customLogger | cache

The configuration properties in this category define the cache settings for the service that collects custom data to a table (an event which uses the `UACICustomLoggerTableName` event parameter).

threshold

Description

The number of records accumulated before the flushCacheToDB service writes the collected custom data to the database.

Default value

100

insertPeriodInSecs

Description

The number of seconds between forced writes to the database.

Default value

3600

Interact | services | responseHist

The configuration properties in this category define the settings for the service that writes to the response history staging tables.

enableLog

Description

If `true`, enables the service that writes to the response history staging tables. If `false`, no data is written to the response history staging tables.

The response history staging table is defined by the `responseHistoryTable` property for the audience level. The default is `UACI_RHStaging`.

Default value

True

Valid Values

True | False

cacheType

Description

Defines whether the cache is kept in memory or in a file. You can use `External Loader File` only if you configured Interact to use a database loader utility.

If you select `Memory Cache`, use the `cache` category settings. If you select `External Loader File`, use the `fileCache` category settings.

Default value

Memory Cache

Valid Values

Memory Cache | External Loader File

actionOnOrphan

Description

This setting determines what to do with response events that do not have corresponding contact events. If set to `NoAction`, the response event is processed as if the corresponding contact event was posted. If set to `Warning`, the response event is processed as if the corresponding contact event was posted, but a warning message is written into `interact.log`. If set to `Skip`, the response even is not processed, and an error message is written into `interact.log`. The setting that you choose here is effective regardless if response history logging is enabled.

Default value

NoAction

Valid Values

NoAction | Warning | Skip

Interact | services | response Hist | responseTypeCodes

The configuration properties in this category define the settings for the response history service.

New category name

Description

The name of your response type code.

code

Description

The custom code for your response type.

Default value

The custom code added in the UA_UsrResponseType table.

action

Description

The action corresponding to the custom response type code.

The action defined for the event this is posted overrides the action defined here. Therefore, if a logAccept event is posted without responseTypeCode, this event is treated as an acceptance event. If a logAccept event is posted with a responseTypeCode that exists in this configuration, the configured action is used to determine if it is an acceptance event. If a logAccept event is posted with a responseTypeCode that does not exist in this configuration, this event is not treated as an acceptance event. When an event is treated as an acceptance event, the learning statistics are updated accordingly if learning is enabled. Offer expression rules are evaluated if there is one based on the acceptance of this offer.

Default value

None

Valid Values

LogAccept | LogReject | None

Interact | services | responseHist | cache

The configuration properties in this category define the cache settings for the service that collects the response history data.

threshold

Description

The number of records accumulated before the flushCacheToDB service writes the collected response history data to the database.

Default value

100

insertPeriodInSecs

Description

The number of seconds between forced writes to the database.

Default value

3600

Interact | services | responseHist | fileCache

The configuration properties in this category define the cache settings for the service that collects the response history data if you are using a database loader utility.

threshold

Description

The number of records accumulated before Interact writes them to the database.

`responseHist` - The table defined by the `responseHistoryTable` property for the audience level. The default is `UACI_RHStaging`.

Default value

100

insertPeriodInSecs

Description

The number of seconds between forced writes to the database.

Default value

3600

Interact | services | crossSessionResponse

The configuration properties in this category define general settings for the crossSessionResponse service and the xsession process. You only need to configure these settings if you are using Interact cross-session response tracking.

enableLog

Description

If `true`, enables the crossSessionResponse service and Interact writes data to the cross-session response tracking staging tables. If `false`, disables the crossSessionResponse service.

Default value

False

xsessionProcessIntervalInSecs

Description

The number of seconds between runs of the xsession process. This process moves data from the cross-session response tracking staging tables to the response history staging table and the built-in learning module.

Default value

180

Valid Values

An integer greater than zero

purgeOrphanResponseThresholdInMinutes

Description

The number of minutes the crossSessionResponse service waits before marking any responses that do not match contacts in the contact and response history tables.

If a response has no match in the contact and response history tables, after `purgeOrphanResponseThresholdInMinutes` minutes, Interact marks the

response with a value of -1 in the Mark column of the xSessResponse staging table. You can then manually match or delete these responses.

Default value

180

Interact | services | crossSessionResponse | cache

The configuration properties in this category define the cache settings for the service that collects cross-session response data.

threshold

Description

The number of records accumulated before the flushCacheToDB service writes the collected cross-session response data to the database.

Default value

100

insertPeriodInSecs

Description

The number of seconds between forced writes to the XSessResponse table.

Default value

3600

Interact | services | crossSessionResponse | OverridePerAudience | [AudienceLevel] | TrackingCodes | byTreatmentCode

The properties in this section define how cross-session response tracking matches treatment codes to contact and response history.

SQL

Description

This property defines whether Interact uses the System Generated SQL or custom SQL defined in the OverrideSQL property.

Default value

Use System Generated SQL

Valid Values

Use System Generated SQL | Override SQL

OverrideSQL

Description

If you do not use the default SQL command to match the treatment code to the contact and response history, enter the SQL or stored procedure here.

This value is ignored if SQL is set to Use System Generated SQL.

Default value

useStoredProcedure

Description

If set to true, the OverrideSQL must contain a reference to a stored procedure which matches the treatment code to the contact and response history.

If set to false, the OverrideSQL, if used, must be an SQL query.

Default value

false

Valid Values

true | false

Type

Description

The associated TrackingCodeType defined in the UACI_TrackingType table in the runtime environment tables. Unless you revise the UACI_TrackingType table, the Type must be 1.

Default value

1

Valid Values

An integer defined in the UACI_TrackingType table.

Interact | services | crossSessionResponse | OverridePerAudience | [AudienceLevel] | TrackingCodes | byOfferCode

The properties in this section define how cross-session response tracking matches offer codes to contact and response history.

SQL

Description

This property defines whether Interact uses the System Generated SQL or custom SQL defined in the OverrideSQL property.

Default value

Use System Generated SQL

Valid Values

Use System Generated SQL | Override SQL

OverrideSQL

Description

If you do not use the default SQL command to match the offer code to the contact and response history, enter the SQL or stored procedure here.

This value is ignored if SQL is set to Use System Generated SQL.

Default value

useStoredProcedure

Description

If set to true, the OverrideSQL must contain a reference to a stored procedure which matches the offer code to the contact and response history.

If set to false, the OverrideSQL, if used, must be an SQL query.

Default value

false

Valid Values

true | false

Type

Description

The associated TrackingCodeType defined in the UACI_TrackingType table in the runtime environment tables. Unless you revise the UACI_TrackingType table, the Type must be 2.

Default value

2

Valid Values

An integer defined in the UACI_TrackingType table.

Interact | services | crossSessionResponse | OverridePerAudience | [AudienceLevel] | TrackingCodes | byAlternateCode

The properties in this section define how cross-session response tracking matches a user-defined alternate code to contact and response history.

Name

Description

This property defines the name for the alternate code. This must match the Name value in the UACI_TrackingType table in the runtime environment tables.

Default value

OverrideSQL

Description

The SQL command or stored procedure to match the alternate code to the contact and response history by offer code or treatment code.

Default value

useStoredProcedure

Description

If set to true, the OverrideSQL must contain a reference to a stored procedure which matches the alternate code to the contact and response history.

If set to false, the OverrideSQL, if used, must be an SQL query.

Default value

false

Valid Values

true | false

Type

Description

The associated TrackingCodeType defined in the UACI_TrackingType table in the runtime environment tables.

Default value

3

Valid Values

An integer defined in the UACI_TrackingType table.

Interact | services | threadManagement | contactAndResponseHist

The configuration properties in this category define thread management settings for the services which collect data for the contact and response history staging tables.

corePoolSize

Description

The number of threads to keep in the pool, even if they are idle, for collecting the contact and response history data.

Default value

5

maxPoolSize

Description

The maximum number of threads to keep in the pool for collecting the contact and response history data.

Default value

5

keepAliveTimeSecs

Description

When the number of threads is greater than the core, this is the maximum time that excess idle threads will wait for new tasks before terminating for collecting the contact and response history data.

Default value

5

queueCapacity

Description

The size of the queue used by the thread pool for collecting the contact and response history data.

Default value

1000

termWaitSecs

Description

At the shutdown of the runtime server, this is the number of seconds to wait for service threads to complete collecting the contact and response history data.

Default value

5

Interact | services | threadManagement | allOtherServices

The configuration properties in this category define the thread management settings for the services which collect the offer eligibility statistics, event activity statistics, default string usage statistics, and the custom log to table data.

corePoolSize

Description

The number of threads to keep in the pool, even if they are idle, for the services which collect the offer eligibility statistics, event activity statistics, default string usage statistics, and the custom log to table data.

Default value

5

maxPoolSize

Description

The maximum number of threads to keep in the pool for the services which collect the offer eligibility statistics, event activity statistics, default string usage statistics, and the custom log to table data.

Default value

5

keepAliveTimeSecs

Description

When the number of threads is greater than the core, this is the maximum time that excess idle threads wait for new tasks before terminating for the services which collect the offer eligibility statistics, event activity statistics, default string usage statistics, and the custom log to table data.

Default value

5

queueCapacity

Description

The size of the queue used by the thread pool for the services which collect the offer eligibility statistics, event activity statistics, default string usage statistics, and the custom log to table data.

Default value

1000

termWaitSecs

Description

At the shutdown of the runtime server, this is the number of seconds to wait for service threads to complete for the services which collect the offer eligibility statistics, event activity statistics, default string usage statistics, and the custom log to table data.

Default value

5

Interact | services | threadManagement | flushCacheToDB

The configuration properties in this category define the thread management settings for the threads that write collected data in cache to the runtime environment database tables.

corePoolSize

Description

The number of threads to keep in the pool for scheduled threads that write cached data to the data store.

Default value

5

maxPoolSize

Description

The maximum number of threads to keep in the pool for scheduled threads that that write cached data to the data store.

Default value

5

keepAliveTimeSecs

Description

When the number of threads is greater than the core, this is the maximum time that excess idle threads wait for new tasks before terminating for scheduled threads that that write cached data to the data store.

Default value

5

queueCapacity

Description

The size of the queue used by the thread pool for scheduled threads that that write cached data to the data store.

Default value

1000

termWaitSecs

Description

At the shutdown of the runtime server, this is the number of seconds to wait for service threads to complete for scheduled threads that that write cached data to the data store.

Default value

5

Interact | services | threadManagement | eventHandling

The configuration properties in this category define the thread management settings for the services which collect data for event handling.

corePoolSize

Description

The number of threads to keep in the pool, even if they are idle, for collecting event handling data.

Default value

1

maxPoolSize

Description

The maximum number of threads to keep in the pool for the services which collect the event handling data.

Default value

5

keepAliveTimeSecs

Description

When the number of threads is greater than the core, this is the maximum time that excess idle threads wait for new tasks before terminating for collecting the event handling data.

Default value

5

queueCapacity

Description

The size of the queue used by the thread pool for collecting event handling data.

Default value

1000

termWaitSecs

Description

At the shutdown of the runtime server, this is the number of seconds to wait for service threads to complete for the services which collect the event handling data.

Default value

5

Interact | services | configurationMonitor

The configuration properties in this category allow you to enable or disable integration with Interact Advanced Patterns without having to restart Interact real time, and they define the interval for polling the property value that enables the integration.

enable

Description

If `true`, enables the service that refreshes the value of the **Interact | services | eventPattern | advancedPatterns enableAdvancedPatterns** property. If `false`, you must restart Interact real time when you change the value of the **Interact | services | eventPattern | advancedPatterns enableAdvancedPatterns** property.

Default value

False

Valid Values

True | False

refreshIntervalInMinutes

Description

Defines the time interval for polling the value of the **Interact | services | eventPattern | advancedPatterns enableAdvancedPatterns** property.

Default value

5

Interact | cacheManagement

This set of configuration properties defines settings for selecting and configuring each of the supported cache managers that you can use to improve the performance of Interact, such as EHCache, which is built-in to your Interact installationWebSphere eXtreme Scale caching, which is an optional add-on, or another external caching system.

Use the **Interact | cacheManagement | Cache Managers** configuration properties to configure the cache manager you want to use. Use the **Interact | cacheManagement | caches** configuration properties to specify which cache manager Interact should use to improve performance.

Interact | cacheManagement | Cache Managers

The Cache Managers category specifies the parameters for the cache management solutions you plan to use with Interact.

Interact | cacheManagement | Cache Managers | EHCache

The EHCache category specifies the parameters for the EHCache cache management solution, so that you can customize it to improve the performance of Interact.

Interact | Cache Managers | EHCache | Parameter Data

The configuration properties in this category control how the EHCache cache management system works to improve the performance of Interact.

cacheType

Description

You can configure the Interact runtime servers in a server group to use a multicast address for sharing cache data. This is referred to as a *distributed cache*. The cacheType parameter specifies whether you are using the built-in EHCache caching mechanism in **local** (stand-alone) mode or **distributed** (as with a runtime server group).

Note:

If you select **Distributed** for the cacheType, all of the servers sharing the cache must be part of the same, single server group. You must also enable multicast to work between all members of a server group.

Default value

Local

Valid Values

Local | Distributed

multicastIPAddress

Description

If you specify that the **cacheType** parameter is "distributed," you are configuring the cache to operate via multicast between all members of an Interact runtime server group. The multicastIPAddress value is the IP address that all the Interact servers for the server group use for listening.

The IP address must be unique across your server groups.

Default value

230.0.0.1

multicastPort

Description

If you specify that the **cacheType** parameter is "distributed," the **multicastPort** parameter indicates the port that all of the Interact servers for the server group use for listening.

Default value

6363

overflowToDisk

Description

The EHCache cache manager manages the session information using available memory. For environments where the session size is large due to a large profile, the number of sessions to be supported in memory may not be large enough to support the customer scenario. For situations where this is the case, EHCache has an optional feature to allow cache information greater than the amount that can be kept in memory to be written temporarily to the hard drive instead.

If you set the **overflowToDisk** property to "yes," each Java virtual machine (JVM) can handle more concurrent sessions than the memory alone would have allowed.

Default value

No

Valid Values

No | Yes

diskStore

Description

When the configuration property **overflowToDisk** is set to `Yes`, this configuration property specifies the disk directory that will hold the cache entries that are overflowed from memory. If this configuration property does not exist or its value is not valid, the disk directory is automatically created in the operating system's default temporary directory.

Default value

None

Valid Values

A directory to which the web application hosting Interact run time has write privileges.

(Parameter)

Description

A template that you can use to create a custom parameter to be used with the cache manager. You can set up any parameter name, and the value it must have.

To create a custom parameter, click ***(Parameter)*** and complete the name and the value you want to assign to that parameter. When you click **Save Changes**, the parameter you have created is added to the list in the Parameter Data category.

Default value

None

Interact | cacheManagement | Cache Managers | Extreme Scale

The Extreme Scale category specifies the parameters for the adapter to use the WebSphere eXtreme Scale cache management solution, so that you can customize it to improve the performance of Interact.

ClassName

Description

The fully-qualified name of the class that connects Interact to the WebSphere eXtreme Scale server. It must be `com.unicacorp.interact.cache.extremescale.ExtremeScaleCacheManager`.

Default value

`com.unicacorp.interact.cache.extremescale.ExtremeScaleCacheManager`

ClassPath

Description

The URI of the location of the file `interact_wxs_adapter.jar`, such as `file:///IBM/IMS/Interact/lib/interact_wxs_adapter.jar` or `file:///C:/IBM/IMS/Interact/lib/interact_wxs_adapter.jar`. However, if this jar file is already included in the class path of the hosting application server, this field should be left blank.

Default value

Blank

Interact | Cache Managers | Extreme Scale | Parameter Data

The configuration properties in this category control the WebSphere eXtreme Scale adapter that is optionally included with your Interact installation. These settings must be configured for each Interact run time server that is acting as a client to the eXtreme Scale server grid.

catalogPropertyFile

Description

The URI of the location of the property file used to start the WebSphere eXtreme Scale catalog server. If the Extreme Scale Adapter is used to start the catalog server, this property must be set. Otherwise, it will not be used.

Default value

`file:///C:/depot/Interact/dev/main/extremescale/config/catalogServer.props`

containerPropertyFile

Description

The URI of the location of the property file used to start the WebSphere eXtreme Scale container instances. If the included server component is used to start the WebSphere eXtreme Scale container servers, this property must be set. Otherwise, it is not used.

Default value

`file:///C:/depot/Interact/dev/main/extremescale/config/containerServer.props`

deploymentPolicyFile

Description

The URI of the location of the deployment policy file used to start the WebSphere eXtreme Scale catalog server. If the included server component is used to start the WebSphere eXtreme Scale catalog server, this property must be set. Otherwise, it is not used.

Default value

file:///C:/depot/Interact/dev/main/extremescale/config/deployment.xml

objectGridConfigFile

Description

The URI of the location of the object grid configuration file used to start the WebSphere eXtreme Scale catalog server and also the near-cache component that runs together with the Interact run time server in the same Java Virtual Machine (JVM).

Default value

file:///C:/depot/Interact/dev/main/extremescale/config/objectgrid.xml

gridName

Description

The name of the WebSphere eXtreme Scale grid that holds all Interact caches.

Default value

InteractGrid

catalogURLs

Description

A URL containing the host name or IP address and the port on which the WebSphere eXtreme Scale catalog server is listening for connections.

Default value

None

(Parameter)

Description

A template that you can use to create a custom parameter to be used with the cache manager. You can set up any parameter name, and the value it must have.

To create a custom parameter, click ***(Parameter)*** and complete the name and the value you want to assign to that parameter. When you click **Save Changes**, the parameter you have created is added to the list in the Parameter Data category.

Default value

None

Interact | caches

Use this set of configuration properties to specify which supported cache manager you want to use to improve the performance of Interact, such as Ehcache or WebSphere eXtreme Scale caching, and to configure specific cache properties for the runtime server you are configuring.

This includes the caches for storing session data, event pattern states, and segmentation results. By adjusting those settings, you can specify which cache solution to use for each type of caching, and you can specify individual settings to control how the cache works.

Interact | cacheManagement | caches | InteractCache

The InteractCache category configures the caching for all session objects, including the profile data, segmentation results, most recently delivered treatments, parameters passed through API methods, and other objects used by the Interact run time.

The InteractCache category is required for Interact to work properly.

The InteractCache category can also be configured through an external EHCache configuration for settings that are not supported in **Interact | cacheManagement | Caches**. If you use EHCache, you must ensure that InteractCache is configured properly.

CacheManagerName

Description

The name of the cache manager that handles the Interact cache. The value you enter here must be one of the cache managers defined in the **Interact | cacheManagement | Cache Managers** configuration properties, such as `EHCache` or `Extreme Scale`.

Default value

EHCache

Valid Values

Any cache manager defined in the **Interact | cacheManagement | Cache Managers** configuration property.

maxEntriesInCache

Description

The maximum number of session data objects to store in this cache. When the maximum number of session data objects has been reached, and data for an additional session need to be stored, the least-recently used object is deleted.

Default value

100000

Valid Values

Integer greater than 0.

timeoutInSecs

Description

The time in seconds that have elapsed since a session data object has been used or updated that are used to determine when the object is removed from the cache.

Note: If you upgraded from a version prior to 9.1, then you will need to reconfigure `timeoutInSecs` property because the property moved.

Default value

300

Valid Values

Integer greater than 0.

Interact | Caches | Interact Cache | Parameter Data

The configuration properties in this category control the Interact Cache that is automatically used by your Interact installation. These settings must be configured individually for each Interact run time server.

asyncIntervalMillis

Description

The time in millisecond that the cache manager EHCache should wait before it replicates any changes to other Interact run time instances. If the value is not positive, those changes will be replicated synchronously.

This configuration property is not created by default. If you create this property, it is used only when EHCache is the cache manager, and when the ehCache **cacheType** property is set to `distributed`.

Default value

None.

(Parameter)

Description

A template that you can use to create a custom parameter to be used with the Intearct Cache. You can set up any parameter name, and the value it must have.

To create a custom parameter, click ***(Parameter)*** and complete the name and the value you want to assign to that parameter. When you click **Save Changes**, the parameter you have created is added to the list in the Parameter Data category.

Default value

None

Interact | cacheManagement | caches | PatternStateCache

The PatternStateCache category is used to host the states of event patterns and real time offer suppression rules. By default, this cache is configured as a read-through and write-through cache, so that Interact attempts to use the cache first event pattern and offer suppression data. If the requested entry does not exist in the cache, the cache implementation loads it from the data source, through either the JNDI configuration or directly using a JDBC connection.

To use a JNDI connection, Interact connects to an existing data source provider that has been defined through the specified server using the JNDI name, URL, and so on. For a JDBC connection, you must provide a set of JDBC settings that include the JDBC driver class name, database URL, and authentication information.

Note that if you define multiple JNDI and JDBC sources, the first enabled JNDI source is used, and if there is no enabled JNDI sources, the first enabled JDBC source is used.

The PatternStateCache category is required for Interact to work properly.

The PatternStateCache category can also be configured through an external EHCache configuration for settings that are not supported in **Interact | cacheManagement | Caches**. If you use EHCache, you must ensure that PatternStateCache is configured properly.

CacheManagerName

Description

The name of the cache manager that handles the Interact pattern state cache. The value you enter here must be one of the cache managers defined in the **Interact | cacheManagement | Cache Managers** configuration properties, such as `EHCache` or `Extreme Scale`.

Default value

EHCache

Valid Values

Any cache manager defined in the **Interact | cacheManagement | Cache Managers** configuration property.

maxEntriesInCache

Description

The maximum number of event pattern states to store in this cache. When the maximum number of event pattern states has been reached, and data for an additional event pattern state need to be stored, the least-recently used object is deleted.

Default value

100000

Valid Values

Integer greater than 0.

timeoutInSecs

Description

Specifies the amount of time, in seconds, for an event pattern state object to time out in the event pattern state cache. When such a state object has been idling in the cache for `timeoutInSecs` number of seconds, it may be ejected from the cache based on the least-recently-used rule. Note that the value of this property should be larger than that defined in the `sessionTimeoutInSecs` property.

Note: If you upgraded from a version prior to 9.1, then you will need to reconfigure `timeoutInSecs` property because the property moved.

Default value

300

Valid Values

Integer greater than 0.

Interact | Caches | PatternStateCache | Parameter Data

The configuration properties in this category control the Pattern State Cache used to host the states of event patterns and real time offer suppression rules.

(Parameter)

Description

A template that you can use to create a custom parameter to be used with the Pattern State Cache. You can set up any parameter name, and the value it must have.

To create a custom parameter, click ***(Parameter)*** and complete the name and the value you want to assign to that parameter. When you click **Save Changes**, the parameter you have created is added to the list in the Parameter Data category.

Default value

None

Interact | cacheManagement | caches | PatternStateCache | loaderWriter

The **loaderWriter** category contains the configuration of the loader that interacts with external repositories for the retrieval and persistence of event patterns.

className

Description

The fully-qualified class name for this loader. This class must comply with the chosen cache manager's requirement.

Default value

```
com.unicacorp.interact.cache.ehcache.loaderwriter.
PatternStateEHCacheLoaderWriter
```

Valid Values

A fully-qualified class name.

classPath

Description

The path to the loader's class file. If you leave this value blank or the entry is invalid, the class path used for running Interact is used.

Default value

None

Valid Values

A valid class path.

writeMode

Description

Specifies the mode for the writer to persist the new or updated event pattern states in the cache. Valid options are:

- WRITE_THROUGH. Every time there is a new entry or an existing entry is updated, that entry is written into the repositories immediately.

- WRITE_BEHIND. The cache manager waits for some time to collect a number of changes, and then persists them into the repositories in a batch.

Default value

WRITE_THROUGH

Valid Values

WRITE_THROUGH or WRITE_BEHIND.

batchSize

Description

The maximum number of event pattern state objects the writer will persist in a batch. This property is used only when **writeMode** is set to `WRITE_BEHIND`.

Default value

100

Valid Values

Integer value.

maxDelayInSecs

Description

The maximum time in seconds that the cache manager waits before an event pattern state object is persisted. This property is used only when **writeMode** is set to `WRITE_BEHIND`.

Default value

5

Valid Values

Integer value.

Interact | Caches | PatternStateCache | loaderWriter | Parameter Data

The configuration properties in this category control the Pattern State Cache loader.

(Parameter)

Description

A template that you can use to create a custom parameter to be used with the Pattern State Cache loader. You can set up any parameter name, and the value it must have.

To create a custom parameter, click ***(Parameter)*** and complete the name and the value you want to assign to that parameter. When you click **Save Changes**, the parameter you have created is added to the list in the Parameter Data category.

Default value

None

Interact | cacheManagement | caches | PatternStateCache | loaderWriter | jndiSettings

The **jndiSettings** category contains the configuration for the JNDI data source the loader will use to communicate with the backing database. To create a new set of JNDI settings, expand the **jdniSettings** category and click the ***(jndiSetting)*** property.

(jndiSettings)

Note: When the WebSphere Application Server is used, the loaderWriter is not get connected with the **jndiSettings**.

Description

When you click this category, a form appears. To define a JNDI data source, complete the following values:

- **New category name** is the name you want to use to identify this JNDI connection.
- **enabled** lets you indicate whether you want this JNDI connection to be available for use or not. Set this to `True` for new connections.
- **jdniName** is the JNDI name that has already been defined in the data source when it was set up.
- **providerUrl** is the URL to find this JNDI data source. If you leave this field blank, the URL of the web application that hosts the Interact run time is used.
- **Initial context factory** is the fully qualified class name of the initial context factory class for connecting to the JNDI provider. If the web application hosting the Interact run time is used for the **providerUrl**, leave this field blank.

Default value

None.

Interact | cacheManagement | caches | PatternStateCache | loaderWriter | jdbcSettings

The **jdbcSettings** category contains the configuration for the JDBC connections the loader will use to communicate with the backing database. To create a new set of JDBC settings, expand the **jdbcSettings** category and click the ***(jdbcSetting)*** property.

(jdbcSettings)

Description

When you click this category, a form appears. To define a JDBC data source, complete the following values:

- **New category name** is the name you want to use to identify this JDBC connection.
- **enabled** lets you indicate whether you want this JDBC connection to be available for use or not. Set this to `True` for new connections.
- **driverClassName** is the fully-qualified class name of the JDBC driver. This class must exist in the class path configured for starting the hosting cache server.
- **databaseUrl** is the URL to find this JDBC data source.

- **asmUser** is the name of the IBM Marketing Software user that has been configured with the credentials for connecting to the database in this JDBC connection.
- **asmDataSource** the name of IBM Marketing Software data source that has been configured with the credentials for connecting to the database in this JDBC connection.
- **maxConnection** is the maximum number of concurrent connections that are allowed to be made the database in this JDBC connection.

Default value

None.

Interact | triggeredMessage

The configuration properties in this category define settings for all triggered messages and offer channel delivery.

backendProcessIntervalMin

Description

This property defines the time period in minutes that the backend thread loads and processes delayed offer deliveries. This value must be an integer. If the value is zero or negative, the backend process is disabled.

Valid Values

A positive integer

autoLogContactAfterDelivery

Description

If this property is set to true, a contact event is automatically posted as soon as this offer is dispatched or this offer is queued for delayed delivery If this property is set to false, no contact event is automatically posted for the outbound offers. This is the default behavior.

Note:

- If you want to capture additional attributes in the contact history when the outbound message is triggered, you can add the additional custom attributes as columns in the contact history. While posting an event, that would trigger the outbound triggered message, you can pass values for the attributes in the `postEvent` method as the name value parameters
- To parametrize an offer to an outbound channel, you could assign offers in the associated strategy, deploy the channel, personalize the offer, and in the triggered message choose **Automatically select next best offer**.

Valid Values

True | False

waitForFlowchart

Description

This property determines if the flowchart should wait for the currently running segmentation to finish, and the behavior if that wait times out.

DoNotWait: The processing of a triggered message starts regardless if segmentation is currently running or not. However, if segments are used in the eligibility rule and/or NextBestOffer is selected as the offer selection method, the TM execution still waits.

OptionalWait : The processing of a triggered message waits until the currently running segmentation finishes or times out. If the wait times out, a warning is logged, and the processing of this triggered message continues. This is the default.

MandatoryWait: The processing of a triggered message waits until the currently running segmentation finishes or times out. If the wait times out, an error is logged, and the processing of this triggered message aborts.

Valid Values

DoNotWait | OptionalWait | MandatoryWait

Interact | triggeredMessage | offerSelection

The configuration properties in this category define settings for offer selection in triggered messages.

maxCandidateOffers

Description

This property defines the maximum number of eligible offers that the engine returns to get the best offer for delivery. There is a chance that none of those returned eligible offers can be sent based on the selected channel. The more candidate offers there are, the less this case happens. However, more candidate offers can increase processing time.

Valid Values

A positive integer

defaultCellCode

Description

If the delivered offer is the result of evaluating a strategic rule or a table driven record, there is a target cell associated to it, and the information of this cell is used in all the related logging. However, if a list of specific offers are used as the input to the offer selection, no target cell is available. In this case, the value of this configuration setting is used. You must make sure this target cell and its campaign are included in the deployment. The easiest method to achieve this is to add the cell into a deployed strategy.

Interact | triggeredMessage | dispatchers

The configuration properties in this category define settings for all dispatchers in triggered messages.

dispatchingThreads

Description

This property defines the number of threads the engine uses to asynchronously call the dispatchers. If the value is 0 or a negative number, the invocation of dispatchers is synchronous. The default value is 0.

Valid Values

An integer

Interact | triggeredMessage | dispatchers | <dispatcherName>

The configuration properties in this category define settings for a specific dispatcher in triggered messages.

category name

Description

This property defines the name of this dispatcher. The name must be unique among all dispatchers.

type

Description

This property defines the disptacher type.

Valid Values

InMemoryQueue | JMSQueue | Custom

Note: If you use JMSQueue or Custom, to integrate Interact with IBM MQ, Interact runtime must be on appserver with JDK 1.7. For WebSphere and WebLogic, it is recommended to use the latest supplied JDK fix pack version.

JMSQueue only supports WebLogic. You cannot use JMSQueue if you use WebSphere Application Server.

className

Description

This property defines the fully qualified class name of this dispatcher implementation. If the type is InMemoryQueue the value should be empty. If the type is custom, this setting must have the value `com.unicacorp.interact.eventhandler.triggeredmessage.dispatchers.IBMMQDispatcher`.

classPath

Description

This property defines the URL to the JAR file that includes the implementation of this dispatcher.

If the type is custom, this setting much have the value `file://<Interact_HOME>/lib/interact_ibmmqdispatcher.jar;file://<Interact_HOME>/lib/com.ibm.mq.allclient.jar;file://<Interact_HOME>/lib/jms.jar`

Interact | triggeredMessage | dispatchers | <dispatcherName> | Parameter Data

The configuration properties in this category define parameters for a specific dispatcher in triggered messages.

You can choose between three types of dispatchers. InMemoryQueue is the internal dispatcher for Interact. Custom is used for IBM MQ. JMSQueue is used to connect to a JMS provider via JNDI.

cateogry name

Description

This property defines the name of this parameter. The name must be unique among all parameters for that dispatcher.

value

Description

This property defines the parameters, in the format of name value pairs, needed by this dispatcher.

Note: All parameters for trigger messages are case sensitive and should be entered as shown here.

If the type is InMemoryQueue, the following parameter is supported.

- queueCapacity: Optional. The maximum offers that can be waiting in the queue to be dispatched. When specified, this property must be a positive integer. If not specified or invalid, the default value (1000) is used.

If the type is Custom, the following parameters are supported.

- providerUrl: <hostname>:port (case sensitive)
- queueManager: The name of the queue manager that was created on the IBM MQ server.
- messageQueueName: The name of the message queue that was created on the IBM MQ server.
- enableConsumer: This property must be set to true.
- asmUserforMQAuth: The user name for logging into the server. It is required when the server enforces authentication. Otherwise, it should not be specified.
- authDS: The password associated with the user name for logging into the server. It is required when the server enforces authentication. Otherwise, it should not be specified.

If the type is JMSQueue, the following parameter is supported.

- providerUrl: The URL to the JNDI provider (case sensitive).
- connectionFactoryJNDI: The JNDI name of the JMS connection factory.
- messageQueueJNDI: The JNDI name of the JMS queue where the triggered messages are sent to and retrieved from.
- enableConsumer: Whether a consumer of those triggered messages should be started in Interact. This property must be set to true. If not specified, the default value (false) is used.
- initialContextFactory: The fully qualified name of the JNDI initial context factory class. IF you use WebLogic, the value of this parameter should be `weblogic.jndi.WLInitialContextFactory`.

Interact | triggeredMessage | gateways | <gatewayName>

The configuration properties in this category define settings for a specific gateway in triggered messages.

Interact does not support multiple instances of the same gateway. All of the gateway configuration files should be accessible from every Interact Runtime node. In the case of a distributed setup, ensure that the gateway files are kept at a shared location.

category name

Description

This proerpty defines the name of this gateway. It must be unique among all gateways.

className

Description

This property defines the fully qualified class name of this gateway implementation.

classPath

Description

This property defines the URI of the JAR file that includes the implementation of this gateway. If left empty, the class path of the hosting Interact application is used.

For example in a windows system, if the gateway JAR file is available in the directory, `C:\IBM\EMM\EmailGateway\IBM_Interact_OMO_OutboundGateway_Silverpop_1.0\lib\OMO_OutboundGateway_Silverpop.jar`, the classPath should be `file:///C:/IBM/EMM/EmailGateway/IBM_Interact_OMO_OutboundGateway_Silverpop_1.0/lib/OMO_OutboundGateway_Silverpop.jar`. In a unix system, if the gateway jar file is available in the directory, `/opt/IBM/EMM/EmailGateway/IBM_Interact_OMO_OutboundGateway_Silverpop_1.0/lib/OMO_OutboundGateway_Silverpop.jar`, the classpath should be `file:///opt/IBM/EMM/EmailGateway/IBM_Interact_OMO_OutboundGateway_Silverpop_1.0/lib/OMO_OutboundGateway_Silverpop.jar`.

Interact | triggeredMessage | gateways | <gatewayName> | Parameter Data

The configuration properties in this category define parameters for a specific gateway in triggered messages.

cateogry name

Description

This property defines the name of this parameter. The name must be unique among all parameters for that gateway.

value

Description

This property defines the parameters, in the format of name value pairs, needed by this gateway. For all gateways, the following parameters are supported.

Note: All parameters for trigger messages are case sensitive and should be entered as shown here.

- validationTimeoutMillis: The duration in milliseconds that the validation of an offer through this gateway timeouts. The default value is 500.

- deliveryTimeoutMillis: The duration in milliseconds that the delivery of an offer using this gateway timeouts. The default value is 1000.

Interact | triggeredMessage | channels

The configuration properties in this category define settings for all channels in triggered messages.

type

Description

This property defines the root node for settings related to a specific gateway. Default uses the built in channel selector, which is based on the list of channels defined on in the triggered messages UI. If default is selected, className and classPath values should be left blank. Custom uses the customer implementation of IChannelSelector.

Valid Values

Default | Custom

className

Description

This property defines the fully qualified class name of the customer implementation of channel selector. This setting is required if the type is Custom.

classPath

Description

This property defines the URL to the JAR file that includes the implementation of the customer implementation of channel selector. If left empty, the class path of the hosting Interact application is used.

Interact | triggeredMessage | channels | <channelName>

The configuration properties in this category define settings for a specific channel in triggered messages.

category name

Description

This property defines the name of the channel through which offers are sent. It should match those defined in the design time under **Campaign | partitions | <partition[N]> | Interact | outboundChannels**.

Interact | triggeredMessage | channels | <channelName> | <handlerName>

The configuration properties in this category define settings for a specific handler in triggered messages that is used to sent offers.

category name

Description

This property defines the name of the handler which the channel will use to send offers.

dispatcher

Description

This property defines the name of the dispatcher through which this handler uses send offers to the gateway. It must be one of those defined under **interact | triggeredMessage | dispatchers**.

gateway

Description

This property defines the name of the gateway to which this handler send offers ultimately. It must be one of those defined under **interact | triggeredMessage | gateways**.

mode

Description

This property defines the usage mode of this handler. If Failover is selected, this handler is used only when all the handlers with higher priorities defined within this channel failed to send offers. If Addon is selected, this handler is used no matter if other handlers have successfully sent offers.

priority

Description

This property defines the priority of this handler. The engine first tries to use the handler with the highest priority for sending offers.

Valid Values

Any integer

Default

100

Interact | triggeredMessage | channels | Parameter Data

The configuration properties in this category define parameters for a specific channel in triggered messages.

category name

Description

This property defines the name of this parameter. The name must be unique among all parameters for that channel.

value

Description

This property defines the parameters, in the format of name value pairs, needed by the channel selector.

If you use **Customer Preferred Channels** for your channel, you must create

Interact | activityOrchestrator

The activity orchestrator category specifies the receivers and gateways for your Interact inbound gateway activity.

Use the **Interact | activityOrchestrator | receivers** configuration properties to configure your Interact receivers. Use the **Interact | activityOrchestrator | gateways** configuration properties to configure your gateways to use in Interact.

Interact | activityOrchestrator | gateways

The activity orchestrator gateway category specifies the gateways for your Interact inbound gateway activity.

Category name

Description

The name of your gateway.

className

Description

This property defines the fully qualified class name of this gateway implementation.

classPath

Description

This property defines the URI to the JAR file that includes the implementation of this gateway. If left empty, the class path of the hosting Interact application is used. It is used only when the type is `Custom`.

Interact | activityOrchestrator | gateways | Parameter Data

You can add gateway parameters for your gateway configuration files, such as `OMO-conf_inbound_UBX_interactEventNameMapping` and `OMO-conf_inbound_UBX_interactEventPayloadMapping`.

Interact | activityOrchestrator | receivers

The activity orchestrator receivers category specifies the event receivers for your Interact inbound gateway activity.

Category name

Description

The name of your receiver.

Type

Description

The type of receiver. You can choose between `IBM MQ` and `Custom`. `Custom` requires you to use an implementation of the `iReceiver`.

Enabled

Description

Select `True` to enable the receiver or `false` to disable the receiver.

className

Description

This property defines the fully qualified class name of this receiver implementation. It is used only when the type is `Custom`.

classPath

Description

This property defines the URI to the JAR file that includes the implementation of this receiver. If left empty, the class path of the hosting Interact application is used. It is used only when the type is `Custom`.

Interact | activityOrchestrator | receivers | Parameter Data

You can add receiver parameters, such as queueManager and messageQueueName to define your receiver queue.

Interact design environment configuration properties

This section describes all the configuration properties for Interact design environment.

Campaign | partitions | partition[n] | reports

The **Campaign | partitions | partition[n] | reports** property defines the different types of folders for reports.

offerAnalysisTabCachedFolder

Description

The `offerAnalysisTabCachedFolder` property specifies the location of the folder that contains the specification for bursted (expanded) offer reports listed on the Analysis tab when you reach it by clicking the Analysis link on the navigation pane. The path is specified by using the XPath notation.

Default value

```
/content/folder[@name='Affinium Campaign - Object Specific
Reports']/folder[@name='offer']/folder[@name='cached']
```

segmentAnalysisTabOnDemandFolder

Description

The `segmentAnalysisTabOnDemandFolder` property specifies the location of the folder that contains the segment reports listed on the Analysis tab of a segment. The path is specified by using the XPath notation.

Default value

```
/content/folder[@name='Affinium Campaign - Object Specific
Reports']/folder[@name='segment']/folder[@name='cached']
```

offerAnalysisTabOnDemandFolder

Description

The `offerAnalysisTabOnDemandFolder` property specifies the location of the folder that contains the offer reports listed on the Analysis tab of an offer. The path is specified by using the XPath notation.

Default value

```
/content/folder[@name='Affinium Campaign - Object Specific
Reports']/folder[@name='offer']
```

segmentAnalysisTabCachedFolder

Description

The segmentAnalysisTabCachedFolder property specifies the location of the folder that contains the specification for bursted (expanded) segment reports listed on the Analysis tab when you reach it by clicking the Analysis link on the navigation pane. The path is specified by using the XPath notation.

Default value

```
/content/folder[@name='Affinium Campaign - Object Specific
Reports']/folder[@name='segment']
```

analysisSectionFolder

Description

The analysisSectionFolder property specifies the location of the root folder where report specifications are stored. The path is specified by using the XPath notation.

Default value

```
/content/folder[@name='Affinium Campaign']
```

campaignAnalysisTabOnDemandFolder

Description

The campaignAnalysisTabOnDemandFolder property specifies the location of the folder that contains the campaign reports listed on the Analysis tab of a campaign. The path is specified by using the XPath notation.

Default value

```
/content/folder[@name='Affinium Campaign - Object Specific
Reports']/folder[@name='campaign']
```

campaignAnalysisTabCachedFolder

Description

The campaignAnalysisTabCachedFolder property specifies the location of the folder that contains the specification for bursted (expanded) campaign reports listed on the Analysis tab when you reach it by clicking the Analysis link on the navigation pane. The path is specified by using the XPath notation.

Default value

```
/content/folder[@name='Affinium Campaign - Object Specific
Reports']/folder[@name='campaign']/folder[@name='cached']
```

campaignAnalysisTabEmessageOnDemandFolder

Description

The campaignAnalysisTabEmessageOnDemandFolder property specifies the location of the folder that contains the eMessage reports listed on the Analysis tab of a campaign. The path is specified by using the XPath notation.

Default value

```
/content/folder[@name='Affinium Campaign']/folder[@name='eMessage
Reports']
```

campaignAnalysisTabInteractOnDemandFolder

Description

Report server folder string for Interact reports.

Default value

/content/folder[@name='Affinium Campaign']/folder[@name='Interact Reports']

Availability

This property is applicable only if you install Interact.

interactiveChannelAnalysisTabOnDemandFolder

Description

Report server folder string for Interactive Channel analysis tab reports.

Default value

/content/folder[@name='Affinium Campaign - Object Specific Reports']/folder[@name='interactive channel']

Availability

This property is applicable only if you install Interact.

Campaign | partitions | partition[n] | Interact | contactAndResponseHistTracking

These configuration properties define settings for the Interact contact and response history module.

isEnabled

Description

If set to `yes`, enables the Interact contact and response history module which copies the Interact contact and response history from staging tables in the Interact runtime to the Campaign contact and response history tables. The property `interactInstalled` must also be set to `yes`.

Default value

no

Valid Values

yes | no

Availability

This property is applicable only if you have installed Interact.

runOnceADay

Description

Specifies whether to run the Contact and Response History ETL once a day. If you set this property to `Yes`, the ETL runs during the scheduled interval specified by `preferredStartTime` and `preferredEndTime`.

If ETL takes more than 24 hours to execute, and thus misses the start time for the next day, it will skip that day and run at the scheduled time the following day. For example, if ETL is configured to run between 1AM to 3AM, and the process starts at 1AM on Monday and completes at 2AM on

Tuesday, the next run, originally scheduled for 1AM on Tuesday, will be skipped, and the next ETL will start at 1AM on Wednesday.

ETL scheduling does not account for Daylight Savings Time changes. For example, if ETL scheduled to run between 1AM and 3AM, it could run at 12AM or 2AM when the DST change occurs.

Default value

No

Availability

This property is applicable only if you have installed Interact.

processSleepIntervalInMinutes

Description

The number of minutes the Interact contact and response history module waits between copying data from the Interact runtime staging tables to the Campaign contact and response history tables.

Default value

60

Valid Values

Any integer greater than zero.

Availability

This property is applicable only if you have installed Interact.

preferredStartTime

Description

The preferred time to start the daily ETL process. This property, when used in conjunction with the preferredEndTime property, sets up the preferred time interval during which you want the ETL to run. The ETL will start during the specified time interval and will process at most the number of records specified using `maxJDBCFetchBatchSize`. The format is HH:mm:ss AM or PM, using a 12-hour clock.

Default value

12:00:00 AM

Availability

This property is applicable only if you have installed Interact.

preferredEndTime

Description

The preferred time to complete the daily ETL process. This property, when used in conjunction with the preferredStartTime property, sets up the preferred time interval during which you want the ETL to run. The ETL will start during the specified time interval and will process at most the number of records specified using `maxJDBCFetchBatchSize`. The format is HH:mm:ss AM or PM, using a 12-hour clock.

Default value

2:00:00 AM

Availability

This property is applicable only if you have installed Interact.

purgeOrphanResponseThresholdInMinutes

Description

The number of minutes the Interact contact and response history module waits before purging responses with no corresponding contact. This prevents logging responses without logging contacts.

Default value

180

Valid Values

Any integer greater than zero.

Availability

This property is applicable only if you have installed Interact.

maxJDBCInsertBatchSize

Description

The maximum number of records of a JDBC batch before committing the query. This is not the max number of records that the Interact contact and response history module processes in one iteration. During each iteration, the Interact contact and response history module processes all available records from the staging tables. However, all those records are broken into `maxJDBCInsertSize` chunks.

Default value

1000

Valid Values

Any integer greater than zero.

Availability

This property is applicable only if you have installed Interact.

maxJDBCFetchBatchSize

Description

The maximum number of records of a JDBC batch to fetch from the staging database. You may need to increase this value to tune the performance of the contact and response history module.

For example, to process 2.5 million contact history records a day, you should set `maxJDBCFetchBatchSize` to a number greater than 2.5M so that all records for one day will be processed.

You could then set `maxJDBCFetchChunkSize` and `maxJDBCInsertBatchSize` to smaller values (in this example, perhaps to 50,000 and 10,000, respectively). Some records from the next day may be processed as well, but would then be retained until the next day.

Default value

1000

Valid Values

Any integer greater than zero

maxJDBCFetchChunkSize

Description

The maximum number of a JDBC chunk size of data read during ETL (extract, transform, load). In some cases, a chunk size greater than insert size can improve the speed of the ETL process.

Default value

1000

Valid Values

Any integer greater than zero

deleteProcessedRecords

Description

Specifies whether to retain contact history and response history records after they have been processed.

Default value

Yes

completionNotificationScript

Description

Specifies the absolute path to a script to run when the ETL is completed. If you specify a script, five arguments are passed to the completion notification script: start time, end time, total number of CH records processed, total number of RH records processed and status. The start time and end time are numeric values representing number of milliseconds elapsed since 1970. The status argument indicates whether the ETL job was a success or failure. 0 indicates a successful ETL job. 1 indicates a failure and that there are some errors in the ETL job.

Default value

None

fetchSize

Description

Allow you to set the JDBC fetchSize when retrieving records from staging tables.

On Oracle databases especially, adjust the setting to the number of records that the JDBC should retrieve with each network round trip. For large batches of 100K or more, try 10000. Be careful not to use too large a value here, because that will have an impact on memory usage and the gains will become negligible, if not detrimental.

Default value

None

daysBackInHistoryToLookupContact

Description

Limits the records that are searched during response history queries to those within the past specified number of days. For databases with a large number of response history records, this can reduce processing time on queries by limiting the search period to the number of days specified.

The default value of 0 indicates that all records are searched.

Default value

0 (zero)

Campaign | partitions | partition[n] | Interact | contactAndResponseHistTracking | runtimeDataSources | [runtimeDataSource]

These configuration properties define the data source for the Interact contact and response history module.

jndiName

Description

Use the `systemTablesDataSource` property to identify the Java Naming and Directory Interface (JNDI) data source that is defined in the application server (Websphere or WebLogic) for the Interact runtime tables.

The Interact runtime database is the database populated with the `aci_runtime` and `aci_populate_runtime` dll scripts and, for example, contains the following tables (among others): `UACI_CHOfferAttrib` and `UACI_DefaultedStat`.

Default value

No default value defined.

Availability

This property is applicable only if you have installed Interact.

databaseType

Description

Database type for the Interact runtime data source.

Default value

SQLServer

Valid Values

SQLServer | Oracle | DB2

Availability

This property is applicable only if you have installed Interact.

schemaName

Description

The name of the schema containing the contact and response history module staging tables. This should be the same as the runtime environment tables.

You do not have to define a schema.

Default value

No default value defined.

Campaign | partitions | partition[n] | Interact | contactAndResponseHistTracking | contactTypeMappings

These configuration properties define the contact type from campaign that maps to a 'contact' for reporting or learning purposes.

contacted

Description

The value assigned to the `ContactStatusID` column of the `UA_DtlContactHist` table in the Campaign system tables for an offer contact. The value must be a valid entry in the `UA_ContactStatus` table. See the *Campaign Administrator's Guide* for details on adding contact types.

Default value

2

Valid Values

An integer greater than zero.

Availability

This property is applicable only if you have installed Interact.

Campaign | partitions | partition[n] | Interact | contactAndResponseHistTracking | responseTypeMappings

These configuration properties define the responses for accept or reject for reporting and learning.

accept

Description

The value assigned to the `ResponseTypeID` column of the `UA_ResponseHistory` table in the Campaign system tables for an accepted offer. The value must be a valid entry in the `UA_UsrResponseType` table. You should assign the `CountsAsResponse` column the value `1`, a response.

See the *Campaign Administrator's Guide* for details on adding response types.

Default value

3

Valid Values

An integer greater than zero.

Availability

This property is applicable only if you have installed Interact.

reject

Description

The value assigned to the `ResponseTypeID` column of the `UA_ResponseHistory` table in the Campaign system tables for a rejected

offer. The value must be a valid entry in the UA_UsrResponseType table. You should assign the CountsAsResponse column the value 2, a reject. See the *Campaign Administrator's Guide* for details on adding response types.

Default value

8

Valid Values

Any integer greater than zero.

Availability

This property is applicable only if you have installed Interact.

Campaign | partitions | partition[n] | Interact | report

These configuration properties define the report names when integrating with Cognos.

interactiveCellPerformanceByOfferReportName

Description

Name for Interactive Cell Performance by Offer report. This name must match the name of this report on the Cognos server.

Default value

Interactive Cell Performance by Offer

treatmentRuleInventoryReportName

Description

Name for Treatment Rule Inventory report. This name must match the name of this report on the Cognos server.

Default value

Channel Treatment Rule Inventory

deploymentHistoryReportName

Description

Name for Deployment History Report report. This name must match the name of this report on the Cognos server

Default value

Channel Deployment History

Campaign | partitions | partition[n] | Interact | learning

These configuration properties enable you to tune the built-in learning module.

confidenceLevel

Description

A percentage indicating how confident you want the learning utility to be before switching from exploration to exploitation. A value of 0 effectively shuts off exploration.

This property is applicable if the Interact > offerserving > optimizationType property for Interact runtime is set to BuiltInLearning only.

Default value

95

Valid Values

An integer between 0 and 95 divisible by 5 or 99.

validateonDeployment

Description

If set to `No`, Interact does not validate the learning module when you deploy. If set to `yes`, Interact validates the learning module when you deploy.

Default value

`No`

Valid Values

Yes | No

maxAttributeNames

Description

The maximum number of learning attributes the Interact learning utility monitors.

This property is applicable if the `Interact > offerserving > optimizationType` property for Interact runtime is set to `BuiltInLearning` only.

Default value

`10`

Valid Values

Any integer.

maxAttributeValues

Description

The maximum number of values the Interact learning module tracks for each learning attribute.

This property is applicable if the `Interact > offerserving > optimizationType` property for Interact runtime is set to `BuiltInLearning` only.

Default value

5

otherAttributeValue

Description

The default name for the attribute value used to represent all attribute values beyond the `maxAttributeValues`.

This property is applicable if the `Interact > offerserving > optimizationType` property for Interact runtime is set to `BuiltInLearning` only.

Default value

Other

Valid Values

A string or number.

Example

If maxAttributeValues is set to 3 and otherAttributeValue is set to other, the learning module tracks the first three values. All of the other values are assigned to the other category. For example, if you are tracking the visitor attribute hair color, and the first five visitors have the hair colors black, brown, blond, red, and gray, the learning utility tracks the hair colors black, brown, and blond. The colors red and gray are grouped under the otherAttributeValue, other.

percentRandomSelection

Description

The percent of the time the learning module presents a random offer. For example, setting percentRandomSelection to 5 means that 5% of the time (5 out of every 100 recommendations), the learning module presents a random offer, independent of the score. Enabling percentRandomSelection overrides the offerTieBreakMethod configuration property. When percentRandomSelection is enabled, this property is set regardless if learning is on or off or if built-in or external learning is used.

Default value

5

Valid Values

Any integer from 0 (which disables the percentRandomSelection feature) up to 100.

recencyWeightingFactor

Description

The decimal representation of a percentage of the set of data defined by the recencyWeightingPeriod. For example, the default value of .15 means that 15% of the data used by the learning utility comes from the recencyWeightingPeriod.

This property is applicable if the Interact > offerserving > optimizationType property for Interact runtime is set to BuiltInLearning only.

Default value

0.15

Valid Values

A decimal value less than 1.

recencyWeightingPeriod

Description

The size in hours of data granted the recencyWeightingFactor percentage of weight by the learning module. For example, the default value of 120

means that the recencyWeightingFactor of the data used by the learning module comes from the last 120 hours.

This property is applicable only if optimizationType is set to builtInLearning.

Default value

120

minPresentCountThreshold

Description

The minimum number of times an offer must be presented before its data is used in calculations and the learning module enters the exploration mode.

Default value

0

Valid Values

An integer greater than or equal to zero.

enablePruning

Description

If set to Yes, the Interact learning module algorithmically determines when a learning attribute (standard or dynamic) is not predictive. If a learning attribute is not predictive, the learning module will not consider that attribute when determining the weight for an offer. This continues until the learning module aggregates learning data.

If set to No, the learning module always uses all learning attributes. By not pruning non-predictive attributes, the learning module may not be as accurate as it could be.

Default value

Yes

Valid Values

Yes | No

Campaign | partitions | partition[n] | Interact | learning | learningAttributes | [learningAttribute]

These configuration properties define the learning attributes.

attributeName

Description

Each attributeName is the name of a visitor attribute you want the learning module to monitor. This must match the name of a name-value pair in your session data.

This property is applicable if the Interact > offerserving > optimizationType property for Interact runtime is set to BuiltInLearning only.

Default value

No default value defined.

Campaign | partitions | partition[n] | Interact | deployment

These configuration properties define deployment settings.

chunkSize

Description

The maximum size of fragmentation in KB for each Interact deployment package.

Default value

500

Availability

This property is applicable only if you have installed Interact.

Campaign | partitions | partition[n] | Interact | serverGroups | [serverGroup]

These configuration properties define server group settings.

serverGroupName

Description

The name of the Interact runtime server group. This is the name that appears on the interactive channel summary tab.

Default value

No default value defined.

Availability

This property is applicable only if you have installed Interact.

Campaign | partitions | partition[n] | Interact | serverGroups | [serverGroup] | instanceURLs | [instanceURL]

These configuration properties define the Interact runtime servers.

instanceURL

Description

The URL of the Interact runtime server. A server group can contain several Interact runtime servers; however, each server must be created under a new category.

Default value

No default value defined.

Example

`http://server:port/interact`

Availability

This property is applicable only if you have installed Interact.

Campaign | partitions | partition[n] | Interact | flowchart

These configuration properties define the Interact runtime environment used for test runs of interactive flowcharts.

serverGroup

Description

The name of the Interact server group Campaign uses to execute a test run. This name must match the category name you create under `serverGroups`.

Default value

No default value defined.

Availability

This property is applicable only if you have installed Interact.

dataSource

Description

Use the `dataSource` property to identify the physical data source for Campaign to use when performing test runs of interactive flowcharts. This property should match the data source defined by the `Campaign > partitions > partitionN > dataSources` property for the test run data source defined for Interact design time.

Default value

No default value defined.

Availability

This property is applicable only if you have installed Interact.

eventPatternPrefix

Description

The `eventPatternPrefix` property is a string value that is prepended to event pattern names to allow them to be used in expressions in Select or Decision processes within interactive flowcharts.

Note that if you change this value, you must deploy global changes in the interactive channel for this updated configuration to take effect.

Default value

`EventPattern`

Availability

This property is applicable only if you have installed Interact.

Campaign | partitions | partition[n] | Interact | whiteList | [AudienceLevel]

These configuration properties define the default cell code under various special circumstances.

DefaultCellCode

Configuration category

`Campaign|partitions|partition[n]|Interact | whiteList | [AudienceLevel] | defaultOffers`

Description

The default cell code Interact uses if you do not define a cell code in the default offers table.

You need to configure these properties only if you are defining global offer assignments.

Default value

No default value defined.

Valid Values

A string that matches the cell code format defined in Campaign

Availability

This property is applicable only if you have installed Interact.

DefaultCellCode

Configuration category

`Campaign|partitions|partition[n]|Interact | whiteList | [AudienceLevel] | offersBySQL`

Description

The default cell code Interact uses for any offer in the OffersBySQL table(s) that has a null value in the cell code column (or if the cell code column is missing altogether. This value must be a valid cell code.

You need to configure these properties only if you are using SQL queries to get a desired set of candidate offers.

Default value

No default value defined.

Valid Values

A string that matches the cell code format defined in Campaign

Availability

This property is applicable only if you have installed Interact.

DefaultCellCode

Configuration category

`Campaign|partitions|partition[n]|Interact | whiteList | [AudienceLevel] | scoreOverride`

Description

The default cell code Interact uses if you do not define a cell code in the score override table.

You need to configure these properties only if you are defining individual offer assignments.

Default value

No default value defined.

Valid Values

A string that matches the cell code format defined in Campaign

Availability

This property is applicable only if you have installed Interact.

Campaign | partitions | partition[n] | Interact | eventPatterns

This configuration property is used when Interact is integrated with Interact Advanced Patterns.

enableAdvancedPatterns

Configuration category

`Campaign|partitions|partition[n]|Interact|eventPatterns`

Description

Set this value to **True** when Interact is integrated with Interact Advanced Patterns.

Default value

False

Availability

This property is applicable only if Interact is integrated with Interact Advanced Patterns.

Campaign | partitions | partition[n] | Interact | Simulator

These configuration properties define the server group you want to use to run API simulations.

serverGroup

Description

Specify the runtime server group that is used to run API simulations.

Default value

defaultServerGroup

Campaign | partitions | partition[n] | server | internal

Properties in this category specify integration settings and the internalID limits for the selected Campaign partition. If your Campaign installation has multiple partitions, set these properties for each partition that you want to affect.

internalIdLowerLimit

Configuration category

`Campaign|partitions|partition[n]|server|internal`

Description

The `internalIdUpperLimit` and `internalIdLowerLimit` properties constrain the Campaign internal IDs to be within the specified range. Note that the values are inclusive: that is, Campaign may use both the lower and upper limit.

Default value

`0` (zero)

internalIdUpperLimit

Configuration category

`Campaign|partitions|partition[n]|server|internal`

Description

The internalIdUpperLimit and internalIdLowerLimit properties constrain the Campaign internal IDs to be within the specified range. The values are inclusive: that is, Campaign may use both the lower and upper limit. If Distributed Marketing is installed, set the value to 2147483647.

Default value

4294967295

eMessageInstalled

Configuration category

Campaign|partitions|partition[n]|server|internal

Description

Indicates that eMessage is installed. When you select Yes, eMessage features are available in the Campaign interface.

The IBM installer sets this property to Yes for the default partition in your eMessage installation. For additional partitions where you installed eMessage, you must configure this property manually.

Default value

No

Valid Values

Yes | No

interactInstalled

Configuration category

Campaign|partitions|partition[n]|server|internal

Description

After installing the Interact design environment, this configuration property should be set to Yes to enable the Interact design environment in Campaign.

If Interact is not installed, set to No. Setting this property to No does not remove Interact menus and options from the user interface. To remove menus and options, you must manually unregister Interact using the configTool utility.

Default value

No

Valid Values

Yes | No

Availability

This property is applicable only if you installed Interact.

MO_UC_integration

Configuration category

Campaign|partitions|partition[n]|server|internal

Description

Enables integration with Marketing Operations for this partition, if the integration is enabled in the **Platform** configuration settings. For more information, see the *IBM Marketing Operations and Campaign Integration Guide*.

Default value

No

Valid Values

Yes | No

MO_UC_BottomUpTargetCells

Configuration category

`Campaign|partitions|partition[n]|server|internal`

Description

For this partition, allows bottom-up cells for Target Cell Spreadsheets, if **MO_UC_integration** is enabled. When set to `Yes`, both top-down and bottom-up target cells are visible, but bottom-up target cells are read-only. For more information, see the *IBM Marketing Operations and Campaign Integration Guide*.

Default value

No

Valid Values

Yes | No

Legacy_campaigns

Configuration category

`Campaign|partitions|partition[n]|server|internal`

Description

For this partition, enables access to campaigns created before Marketing Operations and Campaign were integrated. Applies only if **MO_UC_integration** is set to `Yes`. Legacy campaigns also include campaigns created in Campaign 7.x and linked to Plan 7.x projects. For more information, see the *IBM Marketing Operations and Campaign Integration Guide*.

Default value

No

Valid Values

Yes | No

IBM Marketing Operations - Offer integration

Configuration category

`Campaign|partitions|partition[n]|server|internal`

Description

Enables the ability to use Marketing Operations to perform offer lifecycle management tasks on this partition, if **MO_UC_integration** is enabled for this partition. Offer integration must be enabled in your **Platform**

configuration settings. For more information, see the *IBM Marketing Operations and Campaign Integration Guide*.

Default value

No

Valid Values

Yes | No

UC_CM_integration

Configuration category

`Campaign|partitions|partition[n]|server|internal`

Description

Enables Digital Analytics online segment integration for a Campaign partition. If you set this value to `Yes`, the Select process box in a flowchart provides the option to select **Digital Analytics Segments** as input. To configure the Digital Analytics integration for each partition, choose **Settings > Configuration > Campaign | partitions | partition[n] | Coremetrics**.

Default value

No

Valid Values

Yes | No

numRowsReadToParseDelimitedFile

Configuration category

`Campaign|partitions|partition[n]|server|internal`

Description

This property is used when mapping a delimited file as a user table. It is also used by the Score process box when importing a score output file from IBM SPSS Modeler Advantage Enterprise Marketing Management Edition. To import or map a delimited file, Campaign needs to parse the file to identify the columns, data types (field types), and column widths (field lengths).

The default value of 100 means Campaign examines the first 50 and the last 50 line entries in the delimited file. Campaign then allocates the field length based on the largest value it finds within those entries. In most cases, the default value is sufficient to determine field lengths. However, in very large delimited files, a later field might exceed the estimated length that Campaign computes, which can cause an error during flowchart runtime. Therefore, if you are mapping a very large file, you can increase this value to make Campaign examine more line entries. For example, a value of 200 makes Campaign examine the first 100 line entries and the last 100 line entries of the file.

A value of 0 examines the entire file. Typically, this is necessary only if you are importing or mapping files that have variable data widths of fields which cannot be identified by reading the first and last few lines. Reading the entire file for extremely large files can increase the required processing time for table mapping and Score process box runs.

Default value

100

Valid Values

0 (all lines) or any positive integer

Campaign | monitoring

Properties in the this category specify whether the Operational Monitoring feature is enabled, the URL of the Operational Monitoring server, and caching behavior. Operational Monitoring displays and allows you to control active flowcharts.

cacheCleanupInterval

Description

The `cacheCleanupInterval` property specifies the interval, in seconds, between automatic cleanups of the flowchart status cache.

This property is not available in versions of Campaign earlier than 7.0.

Default value

`600` (10 minutes)

cacheRunCompleteTime

Description

The `cacheRunCompleteTime` property specifies the amount of time, in minutes, that completed runs are cached and display on the Monitoring page.

This property is not available in versions of Campaign earlier than 7.0.

Default value

`4320`

monitorEnabled

Description

The `monitorEnabled` property specifies whether the monitor is turned on.

This property is not available in versions of Campaign earlier than 7.0.

Default value

`FALSE`

Valid values

`TRUE | FALSE`

serverURL

Description

The `Campaign > monitoring > serverURL` property specifies the URL of the Operational Monitoring server. This is a mandatory setting; modify the value if the Operational Monitoring server URL is not the default.

If Campaign is configured to use Secure Sockets Layer (SSL) communications, set the value of this property to use HTTPS. For example: `serverURL=https://`*`host:SSL_port`*`/Campaign/OperationMonitor` where:

- *host* is the name or IP address of the machine on which the web application is installed
- *SSL_Port* is the SSL port of the web application.

Note the `https` in the URL.

Default value

`http://localhost:7001/Campaign/OperationMonitor`

monitorEnabledForInteract

Description

If set to `TRUE`, enables Campaign JMX connector server for Interact. Campaign has no JMX security.

If set to `FALSE`, you cannot connect to the Campaign JMX connector server.

This JMX monitoring is for the Interact contact and response history module only.

Default value

`FALSE`

Valid Values

`TRUE | FALSE`

Availability

This property is applicable only if you have installed Interact.

protocol

Description

Listening protocol for the Campaign JMX connector server, if `monitorEnabledForInteract` is set to yes.

This JMX monitoring is for the Interact contact and response history module only.

Default value

JMXMP

Valid Values

JMXMP | RMI

Availability

This property is applicable only if you have installed Interact.

port

Description

Listening port for the Campaign JMX connector server, if `monitorEnabledForInteract` is set to yes.

This JMX monitoring is for the Interact contact and response history module only.

Default value

2004

Valid Values

An integer between 1025 and 65535.

Availability

This property is applicable only if you have installed Interact.

Contact Optimization configuration properties

This section describes the IBM Contact Optimization configuration properties that are found on the Configuration page.

Campaign | unicaACOListener

These configuration properties are for Contact Optimization listener settings.

serverHost

Description

Set to the host server name for the Contact Optimization installation.

Default value

`localhost`

serverPort

Description

Set to the host server port for the Contact Optimization installation.

Default value

none

useSSL

Description

Set to `True` to connect to the Marketing Platform server by using SSL. Otherwise, set to `False`.

Default value

`False`

Valid Values

`True | False`

keepalive

Description

The number of seconds the Campaign web application waits between sending messages to the Contact Optimization Listener to keep the connection active. Using `keepalive` keeps connections open if your network is configured to close inactive connections.

If set to `0`, the web application does not send any messages.

This `keepalive` property is separate from the Java socket keepAlive.

Default value

`0`

Valid Values

Positive integer

logProcessId

Description

Set to yes to log the ID of the Contact Optimization listener process in the Contact Optimization Listener log (unica_acolsnr.log, in the logs directory of your Contact Optimization installation). Otherwise, set to no.

Default value

yes

Valid Values

yes | no

loggingLevels

Description

You can set the details for the Contact Optimization listener data you log.

This setting affects the Contact Optimization Listener log (unica_acolsnr.log, in the logs directory of your Contact Optimization installation).

Default value

MEDIUM

Valid Values

LOW | MEDIUM | HIGH | ALL

logMaxFileSize

Description

Set this integer to the maximum size for a log file, in bytes. Contact Optimization creates a file after the log file reaches this size. This setting affects the Contact Optimization Listener log (unica_acolsnr.log, in the logs directory of your Contact Optimization installation).

Default value

20485760

enableLogging

Description

Set to True to enable logging. Otherwise, set to False. This setting affects the Contact Optimization Listener log (unica_acolsnr.log, in the logs directory of your Contact Optimization installation).

Default value

True

Valid Values

True | False

logMaxBackupIndex

Description

Set this integer to the number of backup files to store. This setting affects the Contact Optimization Listener log (`unica_acolsnr.log`, in the `logs` directory of your Contact Optimization installation).

Default value

`5`

loggingCategories

Description

You can specify the categories of data you want to log in a comma-separated list. This setting affects the Contact Optimization Listener log (`unica_acolsnr.log`, in the `logs` directory of your Contact Optimization installation).

Default value

`all`

Valid Values

`all | bad_order | cell_access | commands | config | data_errors | dbload | file_access | general | memory | procrun | query | sort | sysquery | table_access | table_io | table_mapping | webproc`

defaultFilePermissions (UNIX only)

Description

The permission level for the generated log files in the numeric format. For example, 777 for read, write, and execute permissions.

Default value

`660` (Owner and Group have read and write access only)

Campaign | unicaACOOptAdmin

These configuration properties define settings for the unicaACOOptAdmin tool.

getProgressCmd

Description

Specifies a value that is used internally. Do not change this value.

Default value

optimize/ext_optimizeSessionProgress.do

Valid Values

`optimize/ext_optimizeSessionProgress.do`

runSessionCmd

Description

Specifies a value that is used internally. Do not change this value.

Default value

optimize/ext_runOptimizeSession.do

Valid Values

optimize/ext_runOptimizeSession.do

loggingLevels

Description

The loggingLevels property controls the amount of detail that is written to the log file for the Contact Optimization command-line tool, which is based on severity. Available levels are LOW, MEDIUM, HIGH, and ALL, with LOW providing the least detail (that is, only the most severe messages are written). The ALL level includes trace messages and is intended primarily for diagnostic purposes.

Default value

HIGH

Valid Values

LOW | MEDIUM | HIGH | ALL

cancelSessionCmd

Description

Specifies a value that is used internally. Do not change this value.

Default value

optimize/ext_stopOptimizeSessionRun.do

Valid Values

optimize/ext_stopOptimizeSessionRun.do

logoutCmd

Description

Specifies a value that is used internally. Do not change this value.

Default value

optimize/ext_doLogout.do

Valid Values

optimize/ext_doLogout.do

getProgressWaitMS

Description

Set this value to the number (integer) of milliseconds between two successive polls to the web application to get progress information. This value is not used if you do not set getProgressCmd.

Default value

1000

Valid Values

An integer greater than zero

Campaign | partitions | partition[n] | Optimize | sessionRunMonitor

These configuration properties are for sessionRunMonitor settings.

progressFetchDelay

Description

Set this integer to the number of milliseconds that the web application waits before it obtains progress information from the listener.

Default value

250

Campaign | partitions | partition[n] | Optimize | MemoryTuning

These configuration properties are for the MemoryTuning settings.

MaxRamUsage

Description

Defines the maximum memory in MB used to cache the contact history. This value must be at least as large as one contact history record.

Default value

128

Campaign | partitions | partition[n] | Optimize | AgentTemplateTables

These configuration properties define template tables for Agent Capacity Optimization.

AgentInfoTemplateTables

Description

Enter a comma-separated list of table names for the Agent Information Template Table. Each table contains the unique identification values (IDs) of agents and their capacity. These tables should be present in the Campaign system database.

Default value

There is no default value defined.

AgentCustomerRelTemplateTables

Description

Enter a comma-separated list of table names for the Agent Customer Relationship Template Table. The Agent Customer Relationship table contains the unique identification values (IDs) of agents and the audience ID of associated customers. This table should be present in the Campaign system database. The audience level of the audience ID must be the same as the audience level of your Contact Optimization session.

Default value

There is no default value defined.

Campaign | partitions | partition[n] | Optimize | userTemplateTables

This property defines the template tables that are used by the PCT and OCT.

tablenames

Description

Enter a comma-separated list of table names for the Contact Optimization template tables. These template tables can be used to add user-specific fields to the proposed contacts table (PCT) or the optimized contacts table (OCT).

Default value

UACO_UserTable

Campaign | partitions | partition[n] | Optimize | TestRun

This property defines the options to use for performing a test run of your Contact Optimization session.

TestRunSamplePercent

Description

The test run sample percentage is the percentage of customers to use from the PCT to perform a test run of your Contact Optimization session.

Default value

10

Valid values

1 - 100

Campaign | partitions | partition[n] | Optimize | AlgorithmTuning

These configuration properties define settings that you can use to tune your optimizations.

MaxAlternativesPerCustomerEvaluated

Description

The maximum number of times Contact Optimization tests combinations of proposed transactions, or alternatives, to find the optimal alternative for a customer.

For example, if the following are true:

- The offers that are associated with a customer in the proposed contacts table (PCT) are A,B,C,D, where the scores for these offers are A=8, B=4, C=2, D=1
- The MaxAlternativesPerCustomerEvaluated property is 5
- A rule of MAX # Offers=3 exists

Then, the alternatives that are tried might be as follows:

- ABC score = 14
- ABD score = 13

- AB score = 12
- ACD score = 11
- AC score = 10

Because the number of alternatives to test might be large, this value limits the effort the core algorithm spends on a customer before Contact Optimization moves to the next customer in the PCT.

Default value

1000

CustomerSampleSize

Description

If your number of customers that are optimized is greater than CustomerSampleSize, Contact Optimization divides the customers into groups of no greater than CustomerSampleSize. Contact Optimization then optimizes each sample group separately. Rules that span across groups, such as a Custom Capacity rule, are still met. Increasing this number might increase optimality but hinder performance.

The most optimal CustomerSampleSize is equal to your number of customers. However, processing a large set of data might take a prohibitive amount of time. By dividing customers into smaller groups for Contact Optimization to process at a time, you can increase performance with minimal loss to optimality.

Default value

1000

Valid Values

Positive integer

MaxIterationsPerCustomerSample

Description

The maximum number of iterations Contact Optimization processes a group of customers. Contact Optimization processes a group of customers until optimality is reached or the number of iterations equals MaxIterationsPerCustomerSample.

Search for the following information in the session log to observe the effect of setting changes for MaxIterationsPerCustomerSample.

- Maximum, minimum, and mean number of iterations per customer chunk
- Maximum, minimum, and mean number of alternatives that are created per customer
- Maximum, minimum, and mean number of alternatives that are tried per customer
- Standard deviation of iterations

Default value

1000

Valid Values

Positive integer

CustomerRandomSeed

Description

The random seed represents the starting point that Contact Optimization uses to select records randomly before Contact Optimization populates sample groups that are defined by the `CustomerSampleSize`. If you have fewer customers than `CustomerSampleSize`, this property has no effect on the optimization.

You might want to change the random seed if you think your current random sample produces highly skewed results.

Default value

`1928374656`

Valid Values

Positive integer

MaxCustomerSampleProcessingThreads

Description

The maximum number of threads Contact Optimization uses to process the optimization algorithms. In general, the higher you set `MaxCustomerSampleProcessingThreads`, the more you might improve performance. However, the performance increase is limited by several factors, including the type and number of optimization rules you use and your hardware. For detailed instructions on tuning your Contact Optimization implementation, contact your IBM representative.

Default value

1

Valid Values

Positive integer

ProcessingThreadQueueSize

Description

The number of threads available to Contact Optimization to use to read a customer sample from the PCT. Increasing the number of threads might improve the performance of a Contact Optimization session. For detailed instructions on tuning your Contact Optimization implementation, contact your IBM representative.

Default value

1

Valid Values

Positive integer

PostProcessingThreadQueueSize

Description

The number of threads available to Contact Optimization to write a customer sample to a staging table for the OCT. Increasing the number of threads might improve the performance of a Contact Optimization session.

For detailed instructions on tuning your Contact Optimization implementation, contact your IBM representative.

Default value

1

Valid Values

Positive integer

EnableMultithreading

Description

If true, Contact Optimization attempts to use multiple threads when processing the optimization algorithms. You can configure the number of threads with the `MaxCustomerSampleProcessingThreads`, `ProcessingThreadQueueSize`, and `PostProcessingThreadQueueSize` configuration properties. If false, Contact Optimization uses a single thread when processing the optimization algorithms.

Default value

`True`

Valid Values

`True | False`

EnableMaxCapacityConsumption

Description

If you get Contact Optimization results that underuse the channel capacities, enable EnableMaxCapacityConsumption to reduce the loss of channel capacity. Then, rerun the Contact Optimization session. If the parameter is set to true, Contact Optimization uses an enhanced algorithm for trying to meet maximum constraints that are set in Cross Customer rules (Min/Max # Offers Capacity rule and Custom capacity rule). However, if this is used, the session run time might increase depending on the data that is provided to the session.

Default value

`False`

Valid Values

`True | False`

EnableBufferingHistoryTransactions

Description

If true, Contact Optimization writes contact history transactions to a file to read during a Contact Optimization session run. If false, Contact Optimization reads from the `UA_ContactHistory` table in the Campaign system tables.

If false, Contact Optimization creates a read lock on the `UA_ContactHistory` table for the length of the Contact Optimization session. This lock might cause attempts to write to the table to fail if you are using a database load utility. If true, Contact Optimization creates a read lock on the table only for the time it takes to write the query to a file.

Default value

False

Valid Values

True | False

MinImprovementPercent

Description

Use this configuration property to stop processing a group of customers when the rate of optimization reaches a specified level. The MinImprovmentPercent property sets a rate of score improvement, which is measured as a percentage, to continue iterating. The default is zero, which means that there is no limit to the number of iterations possible.

Default value

0.0

UseFutureContacts

Description

If you are not using time periods in any of your optimization rules, you can prevent Contact Optimization from querying the Contact History tables to improve performance. You can control this behavior with the UseFutureContacts configuration property.

If you set UseFutureContacts to false, and the optimization rules for your Contact Optimization session do not use time periods, Contact Optimization does not query the Contact History tables. This setting improves the time that is needed to run the Contact Optimization session. However, if the Contact Optimization session uses time periods, Contact History tables are queried.

If you record potential future contacts in Contact History, you must set UseFutureContacts to true. For example, if you know that you are sending an email communication next week about a special promotion to certain customers, those contacts might already be in the Contact History tables as placeholders. In this case, set UseFutureContacts to true and Contact Optimization always queries the Contact History tables.

Default value

False

Valid Values

True | False

ContinueOnGenerationLoopError

Description

If False, Contact Optimization stops the Contact Optimization session if it is not possible to process a set of customers for the following reasons:

- The outer algorithm cannot satisfy the capacity rules with any of its alternate solutions.
- The core algorithm is not creating alternative solutions.

Contact Optimization logs this condition with the following error:

```
The generation loop was unable to eliminate all slack
and surplus variables
```

If True, Contact Optimization skips all the customers in the set that triggered the generation loop error. Contact Optimization then continues processing the next customer set in the Contact Optimization session. It is possible that Contact Optimization might have violated some rules and generated this result without honoring all rules and data. If Optimize|logging|enableBailoutLogging property is also set to TRUE, the skipped customers are logged to unprocessables_*10-digit-session-ID*.csv in the partition/partition[n]/logs directory in the Contact Optimization installation directory. Customers skipped because of the generation loop error have the reason SkippedOnGenerationLoopError.

See the *Contact Optimization Troubleshooting and Tuning Guide* for details about how to avoid the generation loop error.

Default value

False

Valid Values

True | False

Campaign | partitions | partition[n] | Optimize | Debug

This property defines debug level for processing the PCT.

ExtraVerbose

Description

Set this value to yes to provide detailed logs on the rows that are processed in the proposed contacts table (PCT). By default, all rows are logged if you set this value to yes.

If you do not want processed rows of the PCT to be logged, set this value to no.

Default value

no

Valid Values

yes | no

Campaign | partitions | partition[n] | Optimize | DatabaseTuning

These configuration properties are for tuning the database.

UpdateSessionTableStatistics

Description

The UpdateSessionTableStatistics parameter adds a query for updating statistics of the PCT, RC, and POA tables during a Contact Optimization session run. You can tune this parameter at the session level without affecting other sessions. Keeping up-to-date index statistics can help improve the performance of the query on these tables. This parameter is also present in the global configuration settings of Contact Optimization.

Depending on your database, the method to write a query to update statistics varies.

Use following value to update statistics of DB2 tables:

```
CALL SYSPROC.ADMIN_CMD('RUNSTATS ON TABLE <TABLENAME>')
```

Note: If you use DB2 8 or below, you must write your own custom stored procedure for implementing the functionality that is similar to `SYSPROC.ADMIN_CMD`. Also, you can write your own stored procedure to update statistics and start it through the UpdateSessionTableStatistics parameter at run time to update statistics.

Use the following value to update statistics of Oracle tables:

```
analyze table <TABLE> compute statistics
```

Use the following value to update statistics of SQL Server tables:

```
UPDATE STATISTICS <TABLE>
```

Important: If you run this query, the database user in UA_SYSTEM_TABLES must have privileges to run query, which is mentioned in this UpdateSessionTableStatistics parameter. For the session to run successfully, the correct value must be passed or left blank. If the value passed is incorrect, then the session run fails.

Default value

No default value defined.

AllowTempTables

Description

The AllowTempTables parameter creates temporary tables instead of database views, and populates them during the Contact Optimization session run. Enabling this parameter helps in improving run time performance of the Contact Optimization session run. For more information about this parameter, see *Optimize a transaction query for performance improvements* in the *IBM Contact Optimization Troubleshooting and Tuning Guide*.

Default value

`True`

Valid values

`True | False`

Campaign | partitions | partition[n] | Optimize | logging

This property defines logging settings for Contact Optimization.

logMaxBackupIndex

Description

Set this integer to the number of backup files to store. This effects the Contact Optimization Server log (`unica_acosvr_SESSIONID.log` in the `partitions/partition[n]/logs` directory of your Contact Optimization installation.).

Default value

5

logProcessId

Description

Set to True to log the ID of the Contact Optimization server process in the Contact Optimization Server log (unica_acosvr_*SESSIONID*.log in the partitions/partition[n]/logs directory of your Contact Optimization installation). Otherwise, set to False.

Default value

False

Valid Values

True | False

loggingCategories

Description

You can specify the categories of data you want to log in a comma-separated list. This setting affects the Contact Optimization Server log (unica_acosvr_*SESSIONID*.log in the partitions/partition[n]/logs directory of your Contact Optimization installation.).

Default value

all

Valid Values

all | bad_order | cell_access | commands | config | data_errors | dbload | file_access | general | memory | procrun | query | sort | sysquery | table_access | table_io | table_mapping | webproc

loggingLevels

Description

You can set the details for the server data you log.

This setting affects the Contact Optimization Server log unica_acosvr_*SESSIONID*.log in the partitions/partition[n]/logs directory of your Contact Optimization installation).

Default value

MEDIUM

Valid Values

LOW | MEDIUM | HIGH | ALL

enableBailoutLogging

Description

If set to True, Contact Optimization generates a separate file in comma-separated value (CSV) format. The CSV file contains details of customers Contact Optimization cannot process. Contact Optimization cannot process a customer if either of the following are true:

- Contact Optimization exceeds the limit that is set by MaxAlternativesPerCustomerEvaluated, and no legal alternatives are found for a customer.
- ContinueOnGenerationLoopError is set to True and Contact Optimization encounters a generation loop error.

Each row corresponds to one customer. The first column is the customer ID and the second column is the reason why Contact Optimization was not able to process the customer. The file is named unprocessables_*sessionID*.csv and is in the partitions/partition[n]/logs directory of your Contact Optimization installation.

If set to False, Contact Optimization does not generate a list of customers that cannot be processed.

Default value

False

Valid Values

True | False

logMaxFileSize

Description

Set this integer in bytes to the maximum size for a log file. Contact Optimization creates a file after the log file reaches this size. This setting affects the Contact Optimization Server log (unica_acosvr_*SESSIONID*.log in the partitions/partition[n]/logs directory of your Contact Optimization installation.).

Default value

10485760

enableLogging

Description

Set to True to enable logging. Otherwise, set to False. This setting affects the Contact Optimization Server log (*Optimize_installation_directory*/partitions/partition[n]/logs/unica_acosvr_*SESSIONID*.log).

Default value

True

Valid Values

True | False

defaultFilePermissions (UNIX only)

Description

The permission level for the generated log files in the numeric format. For example, *777* for read, write, and run permissions.

Default value

660 (Owner and Group have read and write access only.)

Distributed Marketing configuration properties

This section describes the Distributed Marketing configuration properties on the configuration page.

Additional configuration properties exist in XML files that are located under the Distributed Marketing installation directory.

Collaborate | navigation

These configuration properties are for navigation settings.

welcomePageURI

Description

The Uniform Resource Identifier of the Distributed Marketing index page. You should not change this value.

Default Value

`affiniumcollaborate.jsp?cat=home`

projectDetailpageURI

Description

The Uniform Resource Identifier of the Distributed Marketing detail page. You should not change this value.

Default Value

`uaprojectservlet?cat=projectabs&projecttype=CORPORATE&projectid=`

seedName

Description

Used internally by the Marketing Operations applications. You should not change this value.

Default Value

`Collaborate`

type

Description

Used internally by the Marketing Operations applications. Do not change this value.

Default Value

`Collaborate`

httpPort

Description

The port number that is used by the application server for connections to the Distributed Marketing application.

Default Value

`7001`

httpsPort

Description

The port number that is used by the application server for secure connections to the Distributed Marketing application.

Default Value

`7001`

serverURL

Description

The URL of the Distributed Marketing installation.

If users access Distributed Marketing with the Chrome browser, use the fully qualified domain name (FQDN) in the URL. If the FQDN is not used, the Chrome browser cannot access the product URLs.

Default Value

`http://localhost:7001/collaborate`

displayName

Description

Used internally.

Default Value

`Distributed Marketing`

timeout_redirection

Description

Timeout URL displays. The Distributed Marketing logout page is displayed if empty.

Default Value

No default value is defined.

Collaborate | UDM Configuration Settings

These configuration properties are for Configuration settings.

serverType

Description

The type of web application server you are using. The valid values are `WEBLOGIC` or `WEBSPHERE`.

Default value

userManagerSyncTime

Description

Time in Milliseconds to sync with Marketing Platform. The default value is equivalent to 3 hours.

Default value

`10800000`

showServerLiveClock

Description

This configuration parameter shows the server's clock and time zone information. If the value for this parameter is set to `True`, a message that contains the server time zone and a live clock that shows the server time is displayed on the Summary and Workflow tabs (in both view and edit modes), the Scheduler pop-up, and post-task pop-up pages of Lists, On-demand Campaigns, and Corporate Campaigns.

Default value

`False`

Valid value

`TRUE | FALSE`

firstMonthInFiscalYear

Description

The first month in the fiscal year. The default is 0 for January.

Default value

`0`

systemUserLoginName

Description

The login name of a Marketing Platform user to be used for system tasks (for example, the system task monitor or the scheduler). IBM strongly recommends that the system user is not a normal Distributed Marketing user.

Default value

`[CHANGE-ME]`

searchModifiedTasksForSummaryFrequencyInSeconds

Description

How often, in seconds, to search for changes in task runs to refresh the Summary tab.

Default value

`10`

collaborateFlowchartStatusPeriod

Description

The period in milliseconds between two flowchart status checks.

Default value

`100000`

collaborateFlowchartStatusPeriodRunning

Description

The period in milliseconds between two flowchart status checks when the flowchart is running.

Default value

2000

enableEditProjectCode

Description

If set to true, you can edit the List Code when on the Summary page of the New List wizard. If set to false, you cannot edit the List Code.

Default value

TRUE

Valid value

TRUE | FALSE

minimumDelayForExecutionMonitoring

Description

Optional. Defines the minimum delay, in seconds, for an execution before it appears on the Flowchart Runs Monitoring page.

Default value

10800

validateAllWizardSteps

Description

Determines whether Distributed Marketing checks required fields on non-visited wizard steps. Use this parameter to change behavior that occurs after you click Finish in the project wizard:

- true: Distributed Marketing checks all required fields on all non-visited wizard steps (except workflow, tracking, attachments) when creating a project by using the wizard. If there are any required fields blank, the wizard jumps to that page and displays an error message.
- false: Distributed Marketing does not check required fields on non-visited wizard steps.

Note: Distributed Marketing automatically checks the current page for blank requirement fields. This parameter controls whether Distributed Marketing checks all pages for blank required fields after you click Finish.

Default value

True

Valid value

TRUE | FALSE

Collaborate | UDM Configuration Settings | Attachment

These configuration properties are for Attachment settings.

collaborateModeForAttachments

Description

Distributed Marketing can get the attachments generated by flowchart execution from the Campaign server through the following modes:

- Directory (the default)
- HTTP
- FTP
- TFTP
- SFTP

Default value

`True`

Valid value

`True | False`

collaborateAttachmentsDIRECTORY_directory

Description

Indicates the address in the Campaign server where Distributed Marketing takes the attachments if the mode is set to `Directory`, the default.

Default value

`\Affinium\Campaign\partitions\partition1`

collaborateAttachmentsDIRECTORY_deletefile

Description

The value `True` indicates that the original files will be deleted after copy. The default is `false` if the mode is set to `Directory`.

Default value

`False`

Valid value

`True | False`

collaborateAttachmentsFTP_server

Description

Indicates the server where Distributed Marketing takes the attachments if the mode is set to `FTP`.

Default value

No default value defined.

collaborateAttachmentsFTP_username

Description

Optional. Indicates the user name to log in on FTP server where Distributed Marketing takes the attachments if the parameter collaborateModeForAttachments is `FTP`.

Default value

No default value defined.

collaborateAttachmentsFTP_password

Description

Optional. Indicates the password to log in on FTP server where Distributed Marketing takes the attachments if the parameter collaborateModeForAttachments is `FTP`.

Default value

No default value defined.

collaborateAttachmentsFTP_account

Description

Optional. Indicates the account to log in on FTP server where Distributed Marketing takes the attachments if the parameter collaborateModeForAttachments is `FTP`.

Default value

No default value defined.

collaborateAttachmentsFTP_directory

Description

Optional. Indicates the directory on the FTP server from where Distributed Marketing takes the attachments if the parameter collaborateModeForAttachments is `FTP`. Accepts the relative path of the directory regarding the FTP default directory from where Distributed Marketing can get the attachments for the Windows operating system.

Default value

No default value defined.

collaborateAttachmentsFTP_transfertype

Description

Optional. Indicates the file transfer type on the FTP server that is used by Distributed Marketing to get the attachments if the parameter collaborateModeForAttachments is `FTP`. The value can be `ASCII` or `BINARY`. The default is `ASCII`.

Default value

No default value defined.

collaborateAttachmentsFTP_deletefile

Description

Optional. The value `True` indicates that the original files will be deleted after copy. The default is `false` if the parameter collaborateModeForAttachments is `HTTP`.

Default value

No default value defined.

collaborateAttachmentsHTTP_url

Description

Indicates the HTTP URL where Distributed Marketing takes the attachments if the parameter collaborateModeForAttachments is `HTTP`.

Default value

No default value defined.

collaborateAttachmentsHTTP_deletefile

Description

Optional. The value `True` indicates that the original files will be deleted after copy. The default is `false` if the parameter collaborateModeForAttachments is `HTTP`.

Default value

No default value defined.

collaborateAttachmentsTFTP_server

Description

Indicates the server where Distributed Marketing takes the attachments if the parameter collaborateModeForAttachments is `TFTP`.

Default value

No default value defined.

collaborateAttachmentsTFTP_port

Description

Optional. Indicates the port where Distributed Marketing takes the attachments if the parameter collaborateModeForAttachments is `TFTP`.

Default value

`69`

collaborateAttachmentsTFTP_transfertype

Description

Optional. Indicates the file transfer type on the server that is used by Distributed Marketing to get the attachments if the parameter collaborateModeForAttachments is `TFTP`. The valid values are `ASCII` or `BINARY`. The default is `ASCII`.

Default value

No default value defined.

collaborateAttachmentsSFTP_server

Description

SFTP server name (or IP).

Default value

No default value defined.

collaborateAttachmentsSFTP_port

Description

Optional. FTP server port.

Default value

22

collaborateAttachmentsSFTP_username

Description

User name to log in to the SFTP server.

Default value

No default value defined.

collaborateAttachmentsSFTP_password

Description

Optional. The SFTP password to log in to the SFTP server. It is used if it is needed by the server, and if usepassword=true.

Default value

No default value defined.

collaborateAttachmentsSFTP_usekey

Description

Optional. Use private key file for authenticate user.

Default value

False

Valid values

True | False

collaborateAttachmentsSFTP_keyfile

Description

Optional. SFTP key file name (used if it is needed by the server, and if usekey=true) to log in on SFTP server.

Default value

No default value defined.

collaborateAttachmentsSFTP_keypassphrase

Description

SFTP passphrase to log in on SFTP server. It is used if it is needed by the server, and if usekey=true.

Default value

No default value defined.

collaborateAttachmentsSFTP_knownhosts

Description

Optional. File name for known hosts (used if it is needed by server).

Default value

No default value defined.

collaborateAttachmentsSFTP_directory

Description

Optional. Accepts the relative path of the directory regarding the FTP default directory from where Distributed Marketing can get the attachments for the Windows operating system.

Default value

No default value defined.

collaborateAttachmentsSFTP_deletefile

Description

Optional. Deletes original file after copy, if possible.

Default value

`False`

Valid values

`True | False`

mergeEnabled

Description

Determines whether the merge of documents will be enabled:

- true: the merge is enabled (default).
- false: the merge is disabled.

Default value

`True`

Valid values

`True | False`

mergeFullWritePath

Description

When the merge feature is enabled, this parameter specifies the full path to the merged data file on the local machine.

Default value

`c:/temp`

mergeDataLimitSize

Description

Indicates the upper limit for the size of the date to merge in Microsoft Word. The size is specified in rows (for example, a value of 100 indicates that the merged file cannot contain more than 100 rows). That is, if the number of rows in the file is greater than the value of this parameter, merge is not enabled for this file.

Default value

1000

validateFileUpload

Description

The validateFileUpload property is used to check whether your system validates the file types that are to be uploaded.

Default value

False

Valid values

True | False

upload_allowedFileTypes

Description

Indicates the types of files that can be uploaded in Distributed Marketing.

Default value

doc ppt xls pdf gif jpeg png mpp

upload_fileMaxSize

Description

Indicates the limit on the maximum size of the file that can be uploaded.

Default value

5000000

Collaborate | UDM Configuration Settings | Attachment Folders

These configuration properties are for Attachment Folders settings.

uploadDir

Description

The full path to the Distributed Marketing upload directories. Edit this path to include the full path to the Distributed Marketing upload directories. For example, c:\DistributedMarketing\projectattachments. If you are using UNIX, confirm that Distributed Marketing users have permission to read, write, and run files in this directory.

Default value

projectattachments

taskUploadDir

Description

The full path to the Distributed Marketing task upload directories. Edit this path to include the full path to the Distributed Marketing upload directories. For example, `c:\DistributedMarketing\taskattachments`. If you are using UNIX, confirm that Distributed Marketing users have permission to read, write, and run files in this directory.

Default value

`taskattachments`

Collaborate | UDM Configuration Settings | Campaign Integration

These configuration properties are for Campaign Integration settings.

defaultCampaignPartition

Description

The default Campaign partition. Distributed Marketing uses this parameter if you do not define the <*campaign-partition-id*> tag in a project template file.

Default value

`partition1`

defaultCampaignFolderId

Description

The default Campaign folder ID. Distributed Marketing uses this parameter if you do not define the <*campaign-partition-id*> tag in a project template file.

Default value

`2`

Collaborate | UDM Configuration Settings | Datasource

These configuration properties are for data source settings.

jndiName

Description

Data source name for Distributed Marketing database.

Default value

`collaborateds`

asmJndiName

Description

Data source name for Marketing Platform database, and is used only to synchronize users.

Default value

`UnicaPlatformDS`

Collaborate | UDM Configuration Settings | Flowchart

These configuration properties are for flowchart settings.

enableFlowchartPublishEvent

Description

Specifies whether Distributed Marketing receives events that are sent by Campaign when a flowchart is published.

Default value

`True`

flowchartRepublishOverwriteUserVarPrompt

Description

Specifies whether the User Variable prompt is overwritten when a flowchart is republished.

Default value

`False`

flowchartRepublishOverwriteProcParamPrompt

Description

Specifies whether the Process Parameter prompt is overwritten when a flowchart is republished.

Default value

`False`

flowchartServiceCampaignServicesURL

Description

The URL to the CampaignServices web service that should be used to run flowcharts, get flowchart data, and so on.

Default value

`http://[server-name]:[server-port]/Campaign/services/CampaignServices30Service`

flowchartServiceCampaignServicesTimeout

Description

The number of milliseconds Distributed Marketing waits for communications with the Campaign services before it issues a timeout error.

Default value

`600000`

flowchartServiceNotificationServiceURL

Description

The URL to Distributed Marketing's notification service that receives notifications from Campaign. You must set this parameter for Distributed Marketing to work.

Note: If you use a nonstandard context root, you must specify this parameter.

Default value

`http://[server-name]:[server-port]/collaborate/flowchartRunNotifyServlet`

flowchartServiceCampaignServicesAuthorizationLoginName

Description

A Campaign user with administrative permissions, including access to all data sources, for example, asm_admin.

Default value

`[CHANGE-ME]`

flowchartServiceScheduleServices10Timeout

Description

The number of milliseconds Distributed Marketing waits for communications with the Marketing Platform scheduler before it issues a timeout error.

Default value

`600000`

flowchartServiceScheduleServices10MaxRetries

Description

The number of times Distributed Marketing attempts to connect with the Marketing Platform scheduler before it issues an error.

Default value

`3`

flowchartServiceScheduleServices10RetryPollPeriod

Description

The number of seconds Distributed Marketing waits before it attempts to communicate with the Marketing Platform scheduler again.

Default value

`60`

flowchartServiceScheduleServices10ThrottleType

Description

The types of throttling for scheduled flowchart runs. The valid values are:

- `0`: no throttling (throttle value is ignored)
- `1`: throttle per flowchart instance
- `2`: throttle all flowcharts (default)

Default value

2

flowchartServiceScheduleServices10ThrottleValue

Description

The maximum number of scheduled flowcharts or flowchart instances that can be run at one time.

Default value

10

flowchartServiceSchedulerMonitorPollPeriod

Description

Optional. Defines the approximate time, in seconds, for the scheduler monitor to sleep between polls.

Default value

10

flowchartServiceSchedulerMonitorRemoveSize

Description

Optional. Sets the number of jobs to try to remove from the queue in one shot. The scheduler monitor continues removing events from the event queue in increments that are specified by this value until none are left.

Default value

10

flowchartServiceIsAliveMonitorTimeout

Description

The duration, in seconds, to wait between the start of the flowchart execution and the periodic queries to Campaign of the isAlive monitor.

Default value

900

flowchartServiceIsAliveMonitorMaxRetries

Description

The maximum number of queries that are sent to Campaign by the isAlive monitor before it throws a flowchart run error.

Default value

10

flowchartServiceIsAliveMonitorPollPeriod

Description

The time, in seconds, to wait between queries that are made by the isAlive monitor to Campaign.

Default value

600

Collaborate | UDM Configuration Settings | History

These configuration properties are for history settings.

enableRevisionHistoryPrompt

Description

Ensures that users are prompted to add change comments when they save a project or request or approval.

Default value

False

Valid values

TRUE | FALSE

runHistoryKeep_LIST

Description

Number of run history records to keep for a LIST project. If the value is <=0, Distributed Marketing keeps all run history records.

Default value

-1

runHistoryKeep_LOCAL

Description

Number of run history records (for a List or Campaign flowchart) to keep a local project. If the value is <=0, Distributed Marketing keeps all run history records.

Default value

-1

runHistoryKeep_CORPORATE

Description

Number of run history records (for each run flowchart task) to keep for a corporate project. If the value is <=0, Distributed Marketing keeps all run history records.

Default value

-1

Collaborate | UDM Configuration Settings | Integration Services

These configuration properties are for the integrated services settings.

enableIntegrationServices

Description

This configuration property either enables or disables custom form validation.

Default value

integrationProcedureDefintionPath

Description

This configuration property specifies the location of the `procedure-plugins.xml`.

Default value

`[udm-home]/devkits/integration/examples/src/procedure/procedure-plugins.xml`

integrationProcedureClasspathURL

Description

This configuration property specifies the location of the compiled binary files of the custom validation classes that are defined in the `procedure-plugins.xml` file.

Default value

`file://[udm-home]/devkits/integration/examples/classes/`

Collaborate | UDM Configuration Settings | Listing Pages

These configuration properties are for list pages settings.

listItemsPerPage

Description

Specifies how many items (rows) to be displayed in one list page. This value should be greater than `0`.

Default value

`10`

listPageGroupSize

Description

Specifies the size of visible page numbers in the list navigator in the list page. For example, pages 1-5 is a page group. This value should be greater than `0`.

Default value

`5`

maximumItemsToBeDisplayedInCalendar

Description

The maximum number of objects the system displays on calendars. Use this parameter to restrict users' view of calendars to a specific number of objects. The setting of `0`, the default, indicates that there is no restriction.

Default value

`0`

Collaborate | UDM Configuration Settings | List Manager

These configuration properties are for list manager settings.

listManagerEnabled

Description

Optional. Determines whether marketers can view the List Manager section on the Summary tab:

- true: the List Manager section displays (default)
- false: hides the List Manager section

If you disable the List Manager, you do not need to configure the List Manager configuration files.

Note: The data source to the List Manager table must be active to update the list size after generation.

Default value

`True`

Valid values

`TRUE | FALSE`

listManagerSearchscreenMaxrow

Description

Indicates the maximum number or rows that are returned on the search screen.

Default value

`1000`

listManagerListPageSize

Description

The number of rows that are displayed on a page in the List Manager.

Default value

`20`

listManagerListsMaxrow

Description

The maximum number of rows that are displayed in a list.

Default value

No default value defined.

listManagerResetToValidateIsAllowed_list

Description

By default, when this property is set to `false`, you have the following actions when validating proposed contacts from a List:

- To Validate > Approved
- To Validate > Removed

- Added > Removed
- Approved > Removed
- Removed > Approved

If you set this property to `true`, you can also reset a selection if you made an error with the addition of the following actions:

- Removed > To Validate
- Approved > To Validate

Default value

`False`

Valid values

`TRUE | FALSE`

listManagerResetToValidateIsAllowed_local

Description

By default, when this property is set to `false`, you have the following actions when validating proposed contacts from an On-demand Campaign.

- To Validate > Approved
- To Validate > Removed
- Added > Removed
- Approved > Removed
- Removed > Approved

If you set this property to `true`, you can also reset a selection if you made an error with the addition of the following actions:

- Removed > To Validate
- Approved > To Validate

Default value

`False`

Valid values

`TRUE | FALSE`

listManagerResetToValidateIsAllowed_corporate

Description

By default, when this property is set to `false`, you have the following actions when validating proposed contacts from a Corporate Campaign list:

- To Validate > Approved
- To Validate > Removed
- Added > Removed
- Approved > Removed
- Removed > Approved

If you set this property to `true`, you can also reset a selection if you made an error with the addition of the following actions:

- Removed > To Validate
- Approved > To Validate

Default value

False

Valid values

TRUE | FALSE

Collaborate | UDM Configuration Settings | Lookup Cleanup

These configuration properties are for the lookup cleanup settings.

lookupCleanupMonitorStartDay

Description

Indicates the day when the unused lookup tables or views are automatically cleaned up. The parameter takes weekdays in terms of counts, such as Sunday = 1, Monday = 2. The frequency is weekly.

Default value

2

lookupCleanupMonitorStartTime

Description

Indicates the time when the unused lookup tables or views are automatically cleaned up. The frequency is weekly.

Default value

09:30 am

enableLookupCleanup

Description

When enabled, the lookup monitor runs according to the schedule specified in configuration.

Default value

Disabled

Collaborate | UDM Configuration Settings | Notifications

These configuration properties are for notification settings.

notifyCollaborateBaseURL

Description

The URL for Distributed Marketing. Edit this URL by entering the computer name where you installed Distributed Marketing and the port number you want to use.

Default value

http://[server-name]:[server-port]/collaborate/affiniumcollaborate.jsp

notifyDelegateClassName

Description

Optional. Specifies the fully qualified Java class name of the delegate implementation to be installed by the service.

Default value

No default value defined.

notifyIsDelegateComplete

Description

Indicates that the delegate implementation is complete.

Default value

`true`

Valid values

`TRUE | FALSE`

notifyEventMonitorStartTime

Description

Optional. Time to start the event monitor formatted according to the `java.text.DateFormat` class for the current locale, SHORT version. For example, in US English, the valid string is HH:MM A/PM. The default is set to start immediately after the monitor is created.

Default value

No default value defined.

notifyEventMonitorPollPeriod

Description

Optional. Defines the approximate time, in seconds, for the event monitor to sleep between polls.

Default value

`33`

notifyEventMonitorRemoveSize

Description

Optional. Defines the number of events to try to remove from queue in one shot.

Default value

`10`

Collaborate | UDM Configuration Settings | Notifications | Email

These configuration properties are for email settings.

notifyEmailMonitorJavaMailSession

Description

Optional. Specifies the JNDI name of an existing initialized JavaMail Session to use for email notifications.

Default value

No default value defined.

notifyEmailMonitorJavaMailHost

Description

The machine name or IP address of your organization's SMTP server.

Default value

[none]

notifyEmailMonitorJavaMailProtocol

Description

Optional. Mail server transport protocol to use for email notifications.

Default value

smtp

notifyDefaultSenderEmailAddress

Description

A valid email address for Distributed Marketing to use to send emails when there is otherwise no valid email address available to send notification emails.

Default value

[CHANGE-ME]

notifySenderAddressOverride

Description

Optional. Email address to use for the REPLY-TO and FROM email addresses for notifications. By default, the event number owner's email address is used. If this parameter is not declared or an empty email address is provided, the default addresses are used.

notifyEmailMonitorStartTime

Description

Optional. The time to start the email monitor formatted according to the java.text.DateFormat class for the current locale, SHORT version. For example, in US English, the valid string is HH:MM A/PM. The default is set to start immediately after monitor is created.

Default value

No default value defined.

notifyEmailMonitorPollPeriod

Description

Optional. Defines the approximate time, in seconds, for the email monitor to sleep between polls.

Default value

60

notifyEmailMonitorRemoveSize

Description

Optional. Defines the number of events to try to remove from queue in one shot.

Default value

10

notifyEmailMonitorMaximumResends

Description

Optional. Maximum number of times to try to resend an email after send problems are detected.

Default value

1440

emailMaximumSize

Description

Maximum size, in bytes, of an email.

Default value

2000000

Collaborate | UDM Configuration Settings | Notifications | Project

These configuration properties are for project settings.

notifyProjectAlarmMonitorStartTime

Description

Optional. Time to start the project alarm monitor. If not set, it will start immediately after the monitor is created.

Default value

10:00 pm

notifyProjectAlarmMonitorPollPeriod

Description

Optional. Defines approximate time, in seconds, for the project alarm monitor to sleep between polls.

Default value

86400

notifyProjectAlarmMonitorScheduleStartCondition

Description

Optional. The number of days before a project's start date when Distributed Marketing should start sending start notifications to users. If a project is pending and its start date is within the condition number of days in the future, a PROJECT_SCHEDULED_START notification will be sent out to the appropriate users. If the value is -1, then this condition is not checked for.

Default value

1

notifyProjectAlarmMonitorScheduleEndCondition

Description

Optional. The number of days before a project's end date when Distributed Marketing should start sending notifications to users. If a project is active and its end date is within the condition number of days in the future, a PROJECT_SCHEDULED_END notification will be sent out to the appropriate users. If the value is -1, then this condition is not checked for.

Default value

3

notifyProjectAlarmMonitorScheduleCutoffCondition

Description

Optional. The number of days to start notifying users that a project is scheduled to be closed. If a project is active and its cutoff date is within the condition number of days in the future, a CORPORATE_CAMPAIGN_TO_REVIEW notification will be sent out to the appropriate users. If the value is -1, then this condition is not checked for.

Default value

3

notifyProjectAlarmMonitorTaskScheduledStartCondition

Description

Optional. The number of days before a task's start date when Distributed Marketing should start sending notifications to users. If a task is pending and its start date is within the condition number of days in the future, a TASK_SCHEDULED_START notification will be sent out to the appropriate users. If the value is -1, then this condition is not checked for.

Default value

1

notifyProjectAlarmMonitorTaskScheduledEndCondition

Description

Optional. The number of days before a task's start date when Distributed Marketing should start sending notifications to users that a task did not start. If a task is active and its end date is within the condition number of days in the future, a TASK_SCHEDULED_END notification will be sent out to the appropriate users. If the value is -1, then this condition is not checked for.

Default value

3

notifyProjectAlarmMonitorTaskLateCondition

Description

Optional. The number of days after a task's start date when Distributed Marketing should start sending notifications to users that a task did not start. If a task is pending and its scheduled start date is within the condition number of days in the past, a TASK_LATE notification is sent out to the appropriate users. If the value is -1, then this condition is not checked for.

Default value

3

notifyProjectAlarmMonitorTaskOverdueCondition

Description

Optional. The number of days after a task's end date when Distributed Marketing should be notifying users that a task did not finish. If a task is active and its scheduled end date is within the condition number of days in the past, a TASK_OVERDUE notification is sent out to the appropriate users. If the value is -1, then this condition is not checked for.

Default value

3

notifyProjectAlarmMonitorTaskScheduledMilestoneCondition

Description

Optional. The number of days before a task milestone's start date when Distributed Marketing should start sending notifications to users. If a milestone task is active and its scheduled end date is within the condition number of days in the future, a TASK_SCHEDULED_MILESTONE notification will be sent out to the appropriate users. If the value is -1, then this condition is not checked for.

Default value

1

Collaborate | UDM Configuration Settings | Notifications | System Task

These configuration properties are for system task settings.

systemTaskMonitorStartTime

Description

Optional. The time to start the system task monitor.

- If this parameter contains a value (for example, 11:00 pm), this is the start time for the task monitor to start.
- If this parameter is undefined, the monitor starts immediately after it is created.

Default value

3

systemTaskMonitorPollPeriod

Description

Optional. The duration, in seconds, for the system task monitor to sleep between polls.

Default value

3600

Collaborate | UDM Configuration Settings | Performance

These configuration properties are for performance settings.

commonDataAccessLayerFetchSize

Description

This parameter is a performance optimization that sets the batch size of some performance-sensitive queries. The fetch size is used to determine how many records in the result set are returned to the application at one time.

Default value

500

commonDataAccessLayerMaxResultSetSize

Description

This parameter crops all list page results that are longer than the specified value.

Default value

1000

ssdorSearchResultLimit

Description

The maximum number of rows that are returned by the SSDOR search screen. Increasing this number to a high value may degrade performance.

Default value

500

Collaborate | UDM Configuration Settings | Read Only Lookup Tables

These configuration properties are for lookup table settings.

lookupTableName

Description

Optional. A read-only lookup table name. The lookup table name can include a wildcard, an asterisk (*), at the end of the name. The lookup table is not updated in the Form Editor when a flowchart is republished.

Default value

No default value defined.

New category name

Description

Optional. A template for adding a list of lookup tables to not update during a form merge. When you republish a flowchart, if any attributes in the form are associated with the lookup table, the lookup table and its contents are not updated.

Default value

No default value defined.

Collaborate | UDM Configuration Settings | Reports

These configuration properties are for reports settings.

reportsAnalysisSectionHome

Description

Indicates the home directory for the Analysis Section reports.

Default value

```
/content/folder[@name='Affinium Collaborate']
```

reportsAnalysisTabHome

Description

Indicates the home directory for the object (Corporate Campaign, List, or On-demand Campaign) Analysis Tab reports.

Default value

```
/content/folder[@name='Affinium Collaborate - Object Specific
Reports']
```

reportsAnalysisCorporateSectionHome

Description

Indicates the home directory for the corporate marketer Analysis Section reports.

Default value

```
/content/folder[@name='Affinium Collaborate']
```

reportsAnalysisCorporateTabHome

Description

Indicates the home directory for corporate marketer object (Corporate Campaign, List, or On-Demand Campaign) Analysis Tab reports.

Default value

```
/content/folder[@name='Affinium Collaborate - Object Specific
Reports']/folder[@name='Corporate Marketer']
```

reportsAnalysisFieldMarketerSectionHome

Description

Indicates the home directory for the field marketers Analysis Section reports.

Default value

/content/folder[@name='Affinium Collaborate']/folder[@name='Field Marketer']

reportsAnalysisFieldTabHome

Description

Indicates the home directory for the field marketer object (Corporate Campaign, List, or On-Demand Campaign) Analysis tab reports.

Default value

/content/folder[@name='Affinium Collaborate - Object Specific Reports']/folder[@name='Field Marketer']

Collaborate | UDM Configuration Settings | Siblings

These configuration properties are for sibling settings.

siblingService

Description

Optional. Used to build links to other Distributed Marketing instances to propagate events.

Default value

http://[server-name]:[server-port]/collaborate/services/CollaborateIntegrationServices/1.0

New category name

Description

Optional. A template for giving the URLs specification for the sibling URLs service for the current base instance of Distributed Marketing. Used to build links to other Distributed Marketing instances to propagate events. For example, http://collaborateserver:7001/collaborate/services/CollaborateIntegrationServices/1.0. Do not specify sibling URLs if this configuration file is not the file of the Distributed Marketing base instance.

Default value

No default value defined.

Collaborate | UDM Configuration Settings | Templates

These configuration properties are for template settings.

templatesDir

Description

The directory that contains all your templates. As a best practice, set this to the full path to IBM-Home\DistributedMarketing\templates.

Default value

templates

projectTemplatesFile

Description

The specified file describes the various projects: List, On-demand Campaign, and Corporate Campaign.

Default value

`project_templates.xml`

templateAutoGenerateNameEnabled

Description

Indicates whether the template name for a new template must be generated or not.

Default value

`True`

Valid values

`True | False`

defaultListTableDSName

Description

Used to assign the data source name for templates while importing template if the data source name is not defined.

Default value

`ACC_DEMO`

templateAdminGroup_Name

Description

Specifies multiple groups. Users belonging to these groups have access to template configuration links in Distributed Marketing. Groups with the same name must exist in the Marketing Platform. Multiple groups should be separated by commas.

Default value

`Template Administrators`

Collaborate | UDM Configuration Settings | Workflow

These configuration properties are for workflow settings.

daysInPastRecentTask

Description

How many days in the past Distributed Marketing looks for recent tasks.

Default value

`14`

daysInFutureUpcomingTasks

Description

How many days in the future Distributed Marketing looks for recent tasks.

Default value

14

beginningOfDay

Description

Indicates the beginning hour of the working day with the valid values representing midnight to noon. This setting is used as the denominator when calculating a percentage of task completion in workflows.

Default value

9

Valid values

0 - 12

numberOfHoursPerDay

Description

Indicates the number of hours per day. The default indicates a standard, eight-hour work day. This setting is used as the denominator when calculating a percentage of task completion in workflows.

Default value

8

Valid values

0 - 24

automaticallyRestartFailedRecurrentTasks

Description

Decides whether to automatically restart the failed recurrent tasks. If the value of parameter is set to false, users must manually update the failed task status to Pending either from workflow or from post-task update pop. The schedule picks up only those tasks for runs that are in pending state.

If the value is set to `True`, no manual intervention is required to restart this task.

Default value

`True`

Valid values

`True | False`

projectWorkflowRefreshPeriodInSeconds

Description

System-wide workflow refresh period, in seconds.

Default value

180

IBM SPSS Modeler Advantage Enterprise Marketing Management Edition configuration properties

Properties in this category specify values that are used to configure IBM Marketing Software for single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

See the *IBM Campaign and IBM SPSS Modeler Advantage Enterprise Marketing Management Edition Integration Guide* for complete instructions on setting up single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

SPSS | integration

Properties in this category are used to configure Marketing Platform for single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

Platform user for IBM SPSS account

Description

Enter the login name for the IBM SPSS Modeler Advantage Enterprise Marketing Management Edition account that you created or identified for single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

Default value

`asm_admin`

Availability

This property is used only to configure Marketing Platform for single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

Datasource for IBM SPSS account

Description

Set this property to the name of the data source you created for the system user when you configured single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition. If you used **SPSS_MA_ADMIN_DS** as the data source name, you can retain the default value of this property.

Default value

`SPSS_MA_ADMIN_DS`

Availability

This property is used only to configure Marketing Platform for single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

Is this score only integration

Description

Not supported.

Default value

FALSE

Availability

This property is used only to configure Marketing Platform for single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

SPSS | integration | partitions | partition [n]

The property in this category is used to configure Marketing Platform for single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

Enable IBM SPSS

Description

Set this property to TRUE to enable single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

For each partition where you have users who should have single sign-on, you must use the **SPSS MA EMM Edition | Integration | partitions | partitionTemplate** to create the **enableSPSS** configuration property for that partition. The name of the category you create with the template must exactly match the name of the corresponding Campaign partition. The default partition1 already has the **Enable IBM SPSS** configuration property, so you do not have to use the template to create it.

Default value

FALSE

Availability

This property is used only to configure Marketing Platform for single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

SPSS | navigation

Properties in this category affect IBM SPSS Modeler Advantage Enterprise Marketing Management Edition integration with IBM Campaign. These properties define the location of the Decision Management server and the IBM SPSS Collaboration and Deployment Services server.

IBM SPSS Decision Management Server URL

Description

The URL for the IBM SPSS decision management server. Configure this URL with server name or server IP address followed by the port on which IBM SPSS Modeler Advantage Enterprise Marketing Management Edition is hosted on the server.

Default value

One of the following formats:

- http://<server name>:<port>/DM
- http://<server IP address>:<port>/DM

Valid values

The URL for the IBM SPSS decision management server.

C&DS Server

Description

The name of the IBM SPSS Collaboration and Deployment Services server.

Default value

None

Valid values

Valid server name or server IP address on which IBM SPSS Collaboration and Deployment Services is installed and configured.

C&DS Port

Description

The port where the IBM SPSS Collaboration and Deployment Services server is located.

Default value

None

Valid values

Valid port number on which IBM SPSS Collaboration and Deployment Services is hosted.

Opportunity Detect and Interact Advanced Patterns configuration properties

This section describes the Opportunity Detect and Interact Advanced Patterns configuration properties on the Configuration page.

IBM Opportunity Detect and Interact Advanced Patterns | Navigation

Properties in this category specify values that are used internally to navigate among IBM Marketing Software products.

welcomePageURI

Description

The Uniform Resource Identifier of the IBM Opportunity Detect index page. This value is used internally by IBM Marketing Software applications. Changes to this value are not recommended.

Default value

`/index.jsp`

seedName

Description

Used internally by IBM Marketing Software applications. Changes to this value are not recommended.

Default value

Detect

type

Description

Used internally by IBM Marketing Software applications. Changes to this value are not recommended.

Default value

Detect

httpPort

Description

The port number that is used by the application server for connections to the Opportunity Detect application.

Default value

7001

httpsPort

Description

The port number that is used by the application server for secure connections to the Opportunity Detect application.

Default value

7001

serverURL

Description

The URL of the Opportunity Detect installation. Accepts either the HTTP or HTTPS protocol. If you are on a clustered environment and choose to use ports that are different from the default ports 80 or 443 for your deployment, do not use a port number in the value of this property.

If users access Opportunity Detect with the Chrome browser, use the fully qualified domain name (FQDN) in the URL. If the FQDN is not used, the Chrome browser cannot access the product URLs.

Important: If IBM Marketing Software products are installed in a distributed environment, you must use the machine name rather than an IP address in the navigation URL for all of the applications in the suite.

Default value

[server-url]

logoutURL

Description

Used internally. Changes to this value are not recommended.

IBM Marketing Platform uses this value to call the logout handler of each registered application if the user clicks the logout link in IBM Marketing Software.

serverURLInternal

Description

Used internally. Changes to this value are not recommended.

displayName

Description

Used internally. Changes to this value are not recommended.

Default value

Opportunity Detect

IBM Opportunity Detect and Interact Advanced Patterns | System | Streams Remote Control Web Service

The property in this category specifies the URL for the IBM InfoSphere Streams remote control web service. Opportunity Detect Design Time communicates with Opportunity Detect Run Time over this service.

ServerURL

Description

The person who installs the product sets this property value during installation. The default port number is 8080.

Default value

http://[SRCSHost]:[SRCSPort]/axis2/services/RemoteControl

IBM Opportunity Detect and Interact Advanced Patterns | System | Real Time Connector

The property in this category specifies the URL for the web service used when Interact is integrated with Interact Advanced Patterns or when the Web Service connector is used for input data.

ServerURL

Description

The person who installs the product sets this property value during installation. The default port number is 8282.

Default value

http://[RealTimeConnectorHost]:[RealTimeConnectorPort]/servlets/StreamServlet

IBM Opportunity Detect and Interact Advanced Patterns | System | Monitoring

Properties in this category specify values that affect the monitoring tool.

Poll Interval (In Seconds)

Description

The number of seconds that the monitoring service waits between two successive polls of the Streams server for the statistics. The default is 300 seconds, or 5 minutes.

Default value

300

Retaining Time (In Days)

Description

The number of days the monitoring service should keep the polled data in the database. The default is 10 days. Data that is older than the time specified here is purged.

Default value

10

IBM Opportunity Detect and Interact Advanced Patterns | System | Processing Options

Properties in this category specify values that affect the monitoring tool.

Cache profile records

Description

Opportunity Detect can cache profile data, which provides optimal performance. To enable caching of profile data, set the value of this property to `True`.

If you have very large profile data sets, you might want to retain the default value of this property, which is `False`. This disables caching of profile data and eliminates the out of memory issues that caching large amounts of profile data can cause.

If you change this property value, you must restart your web application server, the Streams instance, and the StreamsRCS service, and redeploy all affected deployments.

Default value

`False`

IBM Opportunity Detect and Interact Advanced Patterns | logging

The property in this category specifies the location of the Opportunity Detect log file.

log4jConfig

Description

The location of the configuration file that Opportunity Detect uses for logging. This value is set automatically during installation, but if you change this path, you must restart the web application server to apply the change.

Default value

`[absolute-path]/conf/detect_log4j.properties`

IBM Interact Advanced Patterns | System | Interact Design Service

The property in this category specifies the URL for the web service that allows Interact to automatically create and deploy advanced patterns when Interact is integrated with Interact Advanced Patterns.

ServerURL

Description

This web service is the integration point between Interact and Interact Advanced Patterns design time. The person who installs the product sets this property value during installation. The default port number is 8181.

Default value

```
http://[InteractServiceHost]:[InteractServicePort]/axis2/services/
InteractDesignService
```

Chapter 22. Customization of stylesheets and images in the IBM Marketing Software user interface

You can customize the appearance of the user interface where most IBM Marketing Software product pages appear. By editing a cascading style sheet and providing your own graphics, you can change many of the images, fonts, and colors in the user interface.

This is sometimes called re-branding, because you can override the IBM logo and color scheme with your company's logo and color scheme.

Stylesheets

The IBM HTML frameset is formatted by a number of cascading style sheets, located in the `css` directory within the `unica.war` file. Several of these stylesheets import a stylesheet named `corporatetheme.css` in the `css\theme` directory. By default, this `corporatetheme.css` file is blank. When you replace this blank file with one that uses your colors and images, you change the appearance of the frameset.

IBM also provides an example `corporatetheme.css` file, in the `css\theme\DEFAULT` directory within the `unica.war` file. This example stylesheet contains all of the specifications that are customizable, along with comments that explain what areas of the frameset each specification affects. You can use this file as a template for making your own changes, as described in the instructions in this section.

Images

Your images can be PNG, GIF, or JPEG format.

IBM uses sprites for some of its buttons and icons. Using sprites reduces the number of HTTP requests going to the server, and can reduce possible flickering. Where IBM uses sprites, the name of the image includes `_sprites`. If you want to replace these images, you should use sprites with the same dimensions, as this requires the fewest modifications to the stylesheet. If you are not familiar with sprites, you can learn about them on the internet.

Preparing your corporate theme

Follow these guidelines to create your corporate theme for the IBM Marketing Software frameset.

1. When you installed Marketing Platform, you may have created an EAR file containing the `unica.war` file, or you may have installed the `unica.war` file. In either case, extract your installed file as necessary to access the files and directories the `unica.war` file contains.
2. Locate the `corporatetheme.css` file, located under in the `css\theme\DEFAULT` directory.
3. See the comments in the `corporatetheme.css` file for details on which area of the framework each stylesheet specification affects.
4. See the images in the `css\theme\img` directory to guide you in creating your images.

© Copyright IBM Corp. 1999, 2018

5. Create your theme in your preferred graphics program and make a note of the image names, fonts, and hexadecimal specifications for the font and background colors.
6. Edit the `corporatetheme.css` file to use your fonts, colors, and images.

Related tasks:

"Applying your corporate theme"

Applying your corporate theme

Follow this procedure to apply your corporate theme to the IBM Marketing Software user interface.

1. Place the images you want to use (for example, your logo, buttons, and icons) in a directory accessible from the machine where Marketing Platform is installed. Refer to the modified `corporatetheme.css` file, created as described in a "Preparing your corporate theme," to determine where to place your images.
2. If Marketing Platform is deployed, undeploy it.
3. When you installed Marketing Platform, you may have created an EAR file containing the `unica.war` file, or you may have installed the `unica.war` file. In either case, do the following.
   - Make a backup of your WAR or EAR file, saving the backup with a different name (for example, `original_unica.war`). This enables you to roll back your changes if necessary.
   - Extract your installed file as necessary to access the files and directories the `unica.war` contains.
4. Place the modified `corporatetheme.css` file, created as described in "Preparing your corporate theme," in the `css\theme` directory.

   This overwrites the blank `corporatetheme.css` file that is already there.
5. Re-create the `unica.war` file, and, if necessary, the EAR file that contained it.
6. Deploy the WAR or EAR file.
7. Clear your browser cache and log in to IBM Marketing Software.

   Your new theme should be applied.

Related tasks:

"Preparing your corporate theme" on page 619

Before you contact IBM technical support

If you encounter a problem that you cannot resolve by consulting the documentation, your company's designated support contact can log a call with IBM technical support. Use these guidelines to ensure that your problem is resolved efficiently and successfully.

If you are not a designated support contact at your company, contact your IBM administrator for information.

Note: Technical Support does not write or create API scripts. For assistance in implementing our API offerings, contact IBM Professional Services.

Information to gather

Before you contact IBM technical support, gather the following information:

- A brief description of the nature of your issue.
- Detailed error messages that you see when the issue occurs.
- Detailed steps to reproduce the issue.
- Related log files, session files, configuration files, and data files.
- Information about your product and system environment, which you can obtain as described in "System information."

System information

When you call IBM technical support, you might be asked to provide information about your environment.

If your problem does not prevent you from logging in, much of this information is available on the About page, which provides information about your installed IBM applications.

You can access the About page by selecting **Help > About**. If the About page is not accessible, check for a `version.txt` file that is located under the installation directory for your application.

Contact information for IBM technical support

For ways to contact IBM technical support, see the IBM Product Technical Support website: (http://www.ibm.com/support/entry/portal/open_service_request).

Note: To enter a support request, you must log in with an IBM account. This account must be linked to your IBM customer number. To learn more about associating your account with your IBM customer number, see **Support Resources > Entitled Software Support** on the Support Portal.

© Copyright IBM Corp. 1999, 2018

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

© Copyright IBM Corp. 1999, 2018

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
B1WA LKG1
550 King Street
Littleton, MA 01460-1250
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating

platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Privacy Policy and Terms of Use Considerations

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. A cookie is a piece of data that a web site can send to your browser, which may then be stored on your computer as a tag that identifies your computer. In many cases, no personal information is collected by these cookies. If a Software Offering you are using enables you to collect personal information through cookies and similar technologies, we inform you about the specifics below.

Depending upon the configurations deployed, this Software Offering may use session and persistent cookies that collect each user's user name, and other personal information for purposes of session management, enhanced user usability, or other usage tracking or functional purposes. These cookies can be disabled, but disabling them will also eliminate the functionality they enable.

Various jurisdictions regulate the collection of personal information through cookies and similar technologies. If the configurations deployed for this Software Offering provide you as customer the ability to collect personal information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for providing notice and consent where appropriate.

IBM requires that Clients (1) provide a clear and conspicuous link to Customer's website terms of use (e.g. privacy policy) which includes a link to IBM's and Client's data collection and use practices, (2) notify that cookies and clear gifs/web beacons are being placed on the visitor's computer by IBM on the Client's behalf along with an explanation of the purpose of such technology, and (3) to the extent required by law, obtain consent from website visitors prior to the placement of cookies and clear gifs/web beacons placed by Client or IBM on Client's behalf on website visitor's devices

For more information about the use of various technologies, including cookies, for these purposes, See IBM's Online Privacy Statement at: http://www.ibm.com/privacy/details/us/en section entitled "Cookies, Web Beacons and Other Technologies."

Printed in USA