

IBM Marketing Platform
Version 9 Release 1
December 2, 2013

Administrator's Guide



Note

Before using this information and the product it supports, read the information in "Notices" on page 515.

This edition applies to version 9, release 1, modification 0 of IBM Marketing Platform and to all subsequent releases and modifications until otherwise indicated in new editions.

© Copyright IBM Corporation 1999, 2013.

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

Contents

Chapter 1. Introduction to the IBM Marketing Platform	1
About IBM EMM security features	1
About configuration management	2
Localization in IBM EMM	2
The common user interface	3
To log in to IBM EMM	3
Chapter 2. User account management.	5
Types of user accounts: internal and external	5
Properties of user accounts	5
Adding a user account	6
Deleting a user account	7
Changing an internal user password expiration date	7
Resetting an internal user password	7
Changing user account properties	7
Changing user system status	8
Adding a user data source	8
Changing a user data source password or login name	8
Deleting a user data source	9
Users window reference	9
The per-user locale preference	11
Setting the user locale preference	11
Synchronization of external users	11
Forcing synchronization of external users	11
Chapter 3. Security management	13
Where to find information on security administration in IBM EMM.	13
About security administration in Marketing Platform	14
About special characters in role and policy names	14
About roles and permissions in Marketing Platform and Campaign	15
The security management process in Marketing Platform	15
Types of groups: internal and external	15
About partitions and security management	16
Pre-configured users and roles	17
Cross-partition administration privileges.	18
Managing internal groups	19
Adding an internal group	19
Adding a subgroup.	19
Deleting a group or subgroup	19
Changing a group or subgroup description.	20
Assigning a group to a partition	20
Adding a user to a group or subgroup	20
Removing a user from a group or subgroup	21
User Groups window reference.	21
Managing user roles and permissions.	23
Creating a role	23
Modifying role permissions	23
Removing a role.	23
Assigning a role to or removing a role from a group	24

Assigning a role to or removing a role from a user	24
Reference: Definition of permission states	25
Reference: Permissions for products that use only basic roles	25
Reference: Permissions for Marketing Platform	26
Reference: Permissions for Interaction History	27
Reference: Permissions for Attribution Modeler	27
Reference: Permissions for Opportunity Detection	27

Chapter 4. Configuration Management	29
About property categories	29
Identifying category types	29
Duplicating categories using templates	29
Deleting categories	30
About property descriptions	31
About refreshing the display	31
About the default user locale preference.	31
Editing property values	31
Navigating to a category	31
Editing property values	32
Duplicating and deleting categories	32
Creating a category from a template	32
Deleting a category	33

Chapter 5. Dashboards	35
Dashboard planning	35
Dashboard audiences	35
User permissions required to view dashboards	35
Pre-defined portlets.	36
Pre-defined portlet availability	36
Pre-assembled dashboards	36
Pre-assembled dashboard availability.	36
Reference: Pre-assembled dashboards.	37
IBM Cognos report performance considerations	38
Scheduling a dashboard report	39
Pre-defined portlet descriptions.	39
Marketing Operations IBM Cognos report portlets	39
Marketing Operations list portlets	40
IBM Cognos report portlets for Campaign	41
Campaign list portlets	42
eMessage IBM Cognos report portlets	42
Interact IBM Cognos report portlet	42
Distributed Marketing list portlets.	43
Contact Optimization list portlets	43
Attribution Modeler IBM Cognos report portlet	43
Interaction History IBM Cognos report portlet.	44
Dashboard setup	44
Permissions required to administer dashboards	44
Dashboard layout	44
Dashboards and partitions	45
Enabling or disabling pre-defined portlets	46
Creating a dashboard that is not pre-assembled	46
Creating a pre-assembled dashboard	46
Adding a pre-defined portlet to a dashboard	47

Assigning or changing a dashboard administrator	47
Removing a portlet from a dashboard	48
Changing the name or properties of a portlet	48
Changing the name or properties of a dashboard	48
Deleting a dashboard	49
Quick link portlets	49
Creating a quick link portlet	49
Custom portlets	50
Custom portlet types and availability	50
Authentication considerations for custom portlets	50
Portlet creation process overview	51
Preparing the URL from an on-premises Digital Analytics for On Premises report	51
Preparing the URL from an IBM Cognos dashboard report	52
Preparing the URL from an IBM Digital Analytics report	52
Preparing the URL from an intranet or internet page	53
Adding a user-created portlet to a dashboard	53
Manage Portlets window reference	54
Create Custom Portlet window reference	54
Dashboard membership administration	54
About dashboard administration tasks	54
Granting or removing dashboard membership	54
Chapter 6. Schedules	57
Scheduler triggers that are sent on success or failure of runs	57
Schedules that depend on completion of multiple runs	58
Schedule triggers that are sent from an external script	59
Scheduler recurrence patterns	60
Time zone support	61
Scheduler throttling	61
Setting up throttling	62
Best practices for setting up schedules	62
Create or edit a schedule window reference	63
What to consider when you use the scheduler with Campaign	64
Difference between the IBM Campaign Schedule process and IBM EMM Scheduler	64
Permissions for scheduling flowcharts	65
Creating a flowchart schedule using default parameters	65
About overriding the default parameters for Campaign flowchart run schedules	66
Schedule notifications	67
Setting up notifications for schedules you create	68
Deleting or modifying notifications for schedules you create	68
Setting up schedule notifications for a group of users	68
Edit Job Notification window reference	68
Schedule management pages	69
Scheduler management window reference	70

Chapter 7. Enabling single sign-on between IBM EMM and IBM Digital Analytics **73**

Setting up single sign-on between IBM EMM and IBM Digital Analytics using automatic user account creation	74
Setting up single sign-on between IBM EMM and IBM Digital Analytics using manual user account creation	75
Configuring your web application server for single sign-on between IBM Digital Analytics and IBM EMM	77

Chapter 8. Integration between IBM EMM and Windows Active Directory **79**

Active Directory integration features	79
Active Directory integration prerequisites	81
Configuration process roadmap (Active Directory integration)	82
Required information	82
Group membership, mapping, and application access	84
Storing directory server credentials in Marketing Platform	84
Setting connection properties	85
Setting LDAP synchronization properties	85
Setting user attributes map properties	86
Mapping LDAP groups to IBM EMM groups	86
Testing synchronization	87
Setting up an Active Directory user with PlatformAdminRole permissions	87
Setting the security mode to Windows Integrated Login	88
Restarting the web application server	88
Configuring Internet Explorer	88
Testing login as an Active Directory user	88

Chapter 9. Integration between IBM EMM and LDAP servers **91**

LDAP integration features	91
LDAP integration prerequisites	93
Configuration process roadmap (LDAP integration)	93
Required information	94
Group membership, mapping, and application access	95
Storing directory server credentials in Marketing Platform	96
Setting connection properties	96
Setting LDAP synchronization properties	97
Setting user attributes map properties	97
Mapping LDAP groups to IBM EMM groups	98
Testing synchronization	99
Setting the security mode to LDAP	99
Restarting the web application server	99
Testing login as an LDAP user	99

Chapter 10. Integration with web access control platforms **101**

SiteMinder integration prerequisites	102
--------------------------------------	-----

Tivoli Access Manager integration prerequisites	105	Create the XML to specify data filters (automatic generation)	142																																																																																																																																																																
How to integrate IBM EMM with a web access control platform	110	Populate the data filter system tables	142																																																																																																																																																																
Configuration process checklist (Web access control integration)	110	Assign users and groups to data filters	142																																																																																																																																																																
Perform LDAP integration	110	Data filter XML reference (automatic generation)	142																																																																																																																																																																
Configure web access control integration in IBM EMM	110	Example: Automatically generating a set of data filters	147																																																																																																																																																																
Restarting the web application server	111	Adding data filters after the initial set has been created	152																																																																																																																																																																
Test web access control synchronization and IBM EMM login	111																																																																																																																																																																		
Chapter 11. Managing alerts and notifications	113																																																																																																																																																																		
Alert and notification subscriptions	113	Chapter 14. Data filter management	153																																																																																																																																																																
Setting system alert and notification subscriptions	113	Configuring email subscriptions	114	Restricting data access through user and group assignments	153			About advanced search	153			Performance	154			Data filter assignments	154			Viewing assigned data filters	154			Assigning users and groups to data filters	154			Removing data filter assignments	155					Chapter 12. SSL implementation	115	Chapter 15. IBM Marketing Platform logs	157	About SSL certificates	115	Client and server roles in IBM EMM	116	About the system log	157	SSL in IBM EMM	117	How to implement SSL in IBM EMM	118	System log configuration	157	Configuration process checklist (SSL)	118			Obtain or create certificates	118	Chapter 16. Configuration process checklists	161	Configure your web application servers for SSL	121	Configuration process checklist (manual specification of data filters)	161	Configuration process roadmap (Active Directory integration)	162	Configure IBM EMM for SSL	122	Configuration process roadmap (LDAP integration)	162	Verifying your SSL configuration	128	Configuration process checklist (Web access control integration)	163	Useful links for SSL	128	Configuration process checklist (SSL)	163					Chapter 13. Data filter configuration	129	Chapter 17. IBM Marketing Platform utilities and SQL scripts	165	About setting up data filters	129	Setting up Marketing Platform utilities on additional machines	166	Data filter associations to restrict user access	129	Marketing Platform utilities	167	Data filter concepts	129	alertConfigTool	167	Two ways to create data filters: automatic generation and manual specification	130	configTool	168	How to set up data filters using manual specification	130	datafilteringScriptTool	172	Configuration process checklist (manual specification of data filters)	130	encryptPasswords	173	Install Marketing Platform	131	partitionTool	174	Plan your data filter criteria (manual generation)	131	populateDb	176	Obtain required information (manual specification)	131	restoreAccess	177	Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141		
Configuring email subscriptions	114	Restricting data access through user and group assignments	153																																																																																																																																																																
		About advanced search	153																																																																																																																																																																
		Performance	154																																																																																																																																																																
		Data filter assignments	154																																																																																																																																																																
		Viewing assigned data filters	154																																																																																																																																																																
		Assigning users and groups to data filters	154																																																																																																																																																																
		Removing data filter assignments	155																																																																																																																																																																
Chapter 12. SSL implementation	115	Chapter 15. IBM Marketing Platform logs	157																																																																																																																																																																
About SSL certificates	115	Client and server roles in IBM EMM	116	About the system log	157	SSL in IBM EMM	117	How to implement SSL in IBM EMM	118	System log configuration	157	Configuration process checklist (SSL)	118			Obtain or create certificates	118	Chapter 16. Configuration process checklists	161	Configure your web application servers for SSL	121	Configuration process checklist (manual specification of data filters)	161	Configuration process roadmap (Active Directory integration)	162	Configure IBM EMM for SSL	122	Configuration process roadmap (LDAP integration)	162	Verifying your SSL configuration	128	Configuration process checklist (Web access control integration)	163	Useful links for SSL	128	Configuration process checklist (SSL)	163					Chapter 13. Data filter configuration	129	Chapter 17. IBM Marketing Platform utilities and SQL scripts	165	About setting up data filters	129	Setting up Marketing Platform utilities on additional machines	166	Data filter associations to restrict user access	129	Marketing Platform utilities	167	Data filter concepts	129	alertConfigTool	167	Two ways to create data filters: automatic generation and manual specification	130	configTool	168	How to set up data filters using manual specification	130	datafilteringScriptTool	172	Configuration process checklist (manual specification of data filters)	130	encryptPasswords	173	Install Marketing Platform	131	partitionTool	174	Plan your data filter criteria (manual generation)	131	populateDb	176	Obtain required information (manual specification)	131	restoreAccess	177	Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																								
Client and server roles in IBM EMM	116	About the system log	157																																																																																																																																																																
SSL in IBM EMM	117	How to implement SSL in IBM EMM	118	System log configuration	157	Configuration process checklist (SSL)	118			Obtain or create certificates	118	Chapter 16. Configuration process checklists	161	Configure your web application servers for SSL	121	Configuration process checklist (manual specification of data filters)	161	Configuration process roadmap (Active Directory integration)	162	Configure IBM EMM for SSL	122	Configuration process roadmap (LDAP integration)	162	Verifying your SSL configuration	128	Configuration process checklist (Web access control integration)	163	Useful links for SSL	128	Configuration process checklist (SSL)	163					Chapter 13. Data filter configuration	129	Chapter 17. IBM Marketing Platform utilities and SQL scripts	165	About setting up data filters	129	Setting up Marketing Platform utilities on additional machines	166	Data filter associations to restrict user access	129	Marketing Platform utilities	167	Data filter concepts	129	alertConfigTool	167	Two ways to create data filters: automatic generation and manual specification	130	configTool	168	How to set up data filters using manual specification	130	datafilteringScriptTool	172	Configuration process checklist (manual specification of data filters)	130	encryptPasswords	173	Install Marketing Platform	131	partitionTool	174	Plan your data filter criteria (manual generation)	131	populateDb	176	Obtain required information (manual specification)	131	restoreAccess	177	Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																														
How to implement SSL in IBM EMM	118	System log configuration	157																																																																																																																																																																
Configuration process checklist (SSL)	118																																																																																																																																																																		
Obtain or create certificates	118	Chapter 16. Configuration process checklists	161																																																																																																																																																																
Configure your web application servers for SSL	121	Configuration process checklist (manual specification of data filters)	161	Configuration process roadmap (Active Directory integration)	162	Configure IBM EMM for SSL	122	Configuration process roadmap (LDAP integration)	162	Verifying your SSL configuration	128	Configuration process checklist (Web access control integration)	163	Useful links for SSL	128	Configuration process checklist (SSL)	163					Chapter 13. Data filter configuration	129	Chapter 17. IBM Marketing Platform utilities and SQL scripts	165	About setting up data filters	129	Setting up Marketing Platform utilities on additional machines	166	Data filter associations to restrict user access	129	Marketing Platform utilities	167	Data filter concepts	129	alertConfigTool	167	Two ways to create data filters: automatic generation and manual specification	130	configTool	168	How to set up data filters using manual specification	130	datafilteringScriptTool	172	Configuration process checklist (manual specification of data filters)	130	encryptPasswords	173	Install Marketing Platform	131	partitionTool	174	Plan your data filter criteria (manual generation)	131	populateDb	176	Obtain required information (manual specification)	131	restoreAccess	177	Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																												
Configuration process checklist (manual specification of data filters)	161	Configuration process roadmap (Active Directory integration)	162																																																																																																																																																																
Configure IBM EMM for SSL	122	Configuration process roadmap (LDAP integration)	162	Verifying your SSL configuration	128	Configuration process checklist (Web access control integration)	163	Useful links for SSL	128	Configuration process checklist (SSL)	163					Chapter 13. Data filter configuration	129	Chapter 17. IBM Marketing Platform utilities and SQL scripts	165	About setting up data filters	129	Setting up Marketing Platform utilities on additional machines	166	Data filter associations to restrict user access	129	Marketing Platform utilities	167	Data filter concepts	129	alertConfigTool	167	Two ways to create data filters: automatic generation and manual specification	130	configTool	168	How to set up data filters using manual specification	130	datafilteringScriptTool	172	Configuration process checklist (manual specification of data filters)	130	encryptPasswords	173	Install Marketing Platform	131	partitionTool	174	Plan your data filter criteria (manual generation)	131	populateDb	176	Obtain required information (manual specification)	131	restoreAccess	177	Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																		
Configuration process roadmap (LDAP integration)	162																																																																																																																																																																		
Verifying your SSL configuration	128	Configuration process checklist (Web access control integration)	163	Useful links for SSL	128	Configuration process checklist (SSL)	163					Chapter 13. Data filter configuration	129	Chapter 17. IBM Marketing Platform utilities and SQL scripts	165	About setting up data filters	129	Setting up Marketing Platform utilities on additional machines	166	Data filter associations to restrict user access	129	Marketing Platform utilities	167	Data filter concepts	129	alertConfigTool	167	Two ways to create data filters: automatic generation and manual specification	130	configTool	168	How to set up data filters using manual specification	130	datafilteringScriptTool	172	Configuration process checklist (manual specification of data filters)	130	encryptPasswords	173	Install Marketing Platform	131	partitionTool	174	Plan your data filter criteria (manual generation)	131	populateDb	176	Obtain required information (manual specification)	131	restoreAccess	177	Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																						
Configuration process checklist (Web access control integration)	163																																																																																																																																																																		
Useful links for SSL	128	Configuration process checklist (SSL)	163					Chapter 13. Data filter configuration	129	Chapter 17. IBM Marketing Platform utilities and SQL scripts	165	About setting up data filters	129	Setting up Marketing Platform utilities on additional machines	166	Data filter associations to restrict user access	129	Marketing Platform utilities	167	Data filter concepts	129	alertConfigTool	167	Two ways to create data filters: automatic generation and manual specification	130	configTool	168	How to set up data filters using manual specification	130	datafilteringScriptTool	172	Configuration process checklist (manual specification of data filters)	130	encryptPasswords	173	Install Marketing Platform	131	partitionTool	174	Plan your data filter criteria (manual generation)	131	populateDb	176	Obtain required information (manual specification)	131	restoreAccess	177	Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																										
Configuration process checklist (SSL)	163																																																																																																																																																																		
Chapter 13. Data filter configuration	129	Chapter 17. IBM Marketing Platform utilities and SQL scripts	165																																																																																																																																																																
About setting up data filters	129	Setting up Marketing Platform utilities on additional machines	166	Data filter associations to restrict user access	129	Marketing Platform utilities	167	Data filter concepts	129	alertConfigTool	167	Two ways to create data filters: automatic generation and manual specification	130	configTool	168	How to set up data filters using manual specification	130	datafilteringScriptTool	172	Configuration process checklist (manual specification of data filters)	130	encryptPasswords	173	Install Marketing Platform	131	partitionTool	174	Plan your data filter criteria (manual generation)	131	populateDb	176	Obtain required information (manual specification)	131	restoreAccess	177	Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																																						
Setting up Marketing Platform utilities on additional machines	166																																																																																																																																																																		
Data filter associations to restrict user access	129	Marketing Platform utilities	167	Data filter concepts	129	alertConfigTool	167	Two ways to create data filters: automatic generation and manual specification	130	configTool	168	How to set up data filters using manual specification	130	datafilteringScriptTool	172	Configuration process checklist (manual specification of data filters)	130	encryptPasswords	173	Install Marketing Platform	131	partitionTool	174	Plan your data filter criteria (manual generation)	131	populateDb	176	Obtain required information (manual specification)	131	restoreAccess	177	Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																																										
Marketing Platform utilities	167																																																																																																																																																																		
Data filter concepts	129	alertConfigTool	167	Two ways to create data filters: automatic generation and manual specification	130	configTool	168	How to set up data filters using manual specification	130	datafilteringScriptTool	172	Configuration process checklist (manual specification of data filters)	130	encryptPasswords	173	Install Marketing Platform	131	partitionTool	174	Plan your data filter criteria (manual generation)	131	populateDb	176	Obtain required information (manual specification)	131	restoreAccess	177	Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																																														
alertConfigTool	167																																																																																																																																																																		
Two ways to create data filters: automatic generation and manual specification	130	configTool	168	How to set up data filters using manual specification	130	datafilteringScriptTool	172	Configuration process checklist (manual specification of data filters)	130	encryptPasswords	173	Install Marketing Platform	131	partitionTool	174	Plan your data filter criteria (manual generation)	131	populateDb	176	Obtain required information (manual specification)	131	restoreAccess	177	Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																																																		
configTool	168																																																																																																																																																																		
How to set up data filters using manual specification	130	datafilteringScriptTool	172	Configuration process checklist (manual specification of data filters)	130	encryptPasswords	173	Install Marketing Platform	131	partitionTool	174	Plan your data filter criteria (manual generation)	131	populateDb	176	Obtain required information (manual specification)	131	restoreAccess	177	Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																																																						
datafilteringScriptTool	172																																																																																																																																																																		
Configuration process checklist (manual specification of data filters)	130	encryptPasswords	173	Install Marketing Platform	131	partitionTool	174	Plan your data filter criteria (manual generation)	131	populateDb	176	Obtain required information (manual specification)	131	restoreAccess	177	Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																																																										
encryptPasswords	173																																																																																																																																																																		
Install Marketing Platform	131	partitionTool	174	Plan your data filter criteria (manual generation)	131	populateDb	176	Obtain required information (manual specification)	131	restoreAccess	177	Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																																																														
partitionTool	174																																																																																																																																																																		
Plan your data filter criteria (manual generation)	131	populateDb	176	Obtain required information (manual specification)	131	restoreAccess	177	Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																																																																		
populateDb	176																																																																																																																																																																		
Obtain required information (manual specification)	131	restoreAccess	177	Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																																																																						
restoreAccess	177																																																																																																																																																																		
Create the XML to specify data filters (manual specification)	132	scheduler_console_client	179	Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																																																																										
scheduler_console_client	179																																																																																																																																																																		
Populate the data filter system tables	132	Marketing Platform SQL scripts	180	Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																																																																														
Marketing Platform SQL scripts	180																																																																																																																																																																		
Assign users and groups to data filters	132	ManagerSchema_DeleteAll.sql	181	Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																																																																																		
ManagerSchema_DeleteAll.sql	181																																																																																																																																																																		
Data filter XML reference (manual specification)	132	ManagerSchema_PurgeDataFiltering.sql	181	Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																																																																																						
ManagerSchema_PurgeDataFiltering.sql	181																																																																																																																																																																		
Example: Manually specifying data filters	135	ManagerSchema_DropAll.sql	181	How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																																																																																										
ManagerSchema_DropAll.sql	181																																																																																																																																																																		
How to set up data filters using automatic specification	139	SQL scripts for creating system tables	182	Configuration process checklists	140			Install Marketing Platform	140			Plan your data filter criteria (automatic generation)	140			Obtain the JDBC driver for your database	141			Obtain required information (automatic generation)	141																																																																																																																																														
SQL scripts for creating system tables	182																																																																																																																																																																		
Configuration process checklists	140																																																																																																																																																																		
Install Marketing Platform	140																																																																																																																																																																		
Plan your data filter criteria (automatic generation)	140																																																																																																																																																																		
Obtain the JDBC driver for your database	141																																																																																																																																																																		
Obtain required information (automatic generation)	141																																																																																																																																																																		

Appendix A. Configuration properties on the Configuration page	185
Marketing Platform configuration properties	185
General Navigation	185

General Data filtering	186
General Password settings	186
General Miscellaneous	189
General Communication Email	189
Platform	191
Platform Scheduler	193
Platform Scheduler Recurrence definitions	194
Platform Scheduler Schedule registrations [Product] [Object type]	196
Platform Scheduler Schedule registrations [Product] [Object type] [Throttling group]	196
Platform Security	197
Platform Security Login method details Windows integrated login	197
Platform Security Login method details LDAP	200
Platform Security Login method details Web access control	202
Platform Security LDAP synchronization	203
Platform Security LDAP synchronization LDAP reference to IBM Marketing Platform group map	211
Platform Notifications	212
IBM Digital Analytics configuration properties	213
Coremetrics	213
Coremetrics Integration partitions partition[n]	213
Interaction History Configuration Properties	214
Interaction History	214
Interaction History navigation	214
Interaction History partitions partition[n] datasource	216
Interaction History partitions partition[n] configuration	217
Interaction History partitions partition[n] CoreMetrics	217
Interaction History partitions partition[n] CampaignAndInteract	222
Interaction History partitions partition[n] eMessage	222
Interaction History partitions partition[n] Reports	223
Attribution Modeler configuration properties	224
Attribution Modeler navigation	224
AttributionModeler AMLListener	225
AttributionModeler partitions partition[n] AMFields	226
Attribution Modeler partitions partition[n] dataSources	227
Attribution Modeler partitions partition[n] dataSources	228
AttributionModeler partitions partition[n] server encoding	228
AttributionModeler partitions partition[n] server logging	229
Attribution Modeler partitions partition[n] AdvancedOptions	231
Report configuration properties	232
Reports Integrations Cognos [version]	232
Reports Schemas [product] [schema name] SQL Configuration	235
Reports Schemas Campaign	236
Reports Schemas Campaign Offer Performance	237
Reports Schemas Campaign [schema name] Columns [Contact Metric]	238
Reports Schemas Campaign [schema name] Columns [Response Metric]	239
Reports Schemas Campaign Campaign Performance	240
Reports Schemas Campaign Campaign Offer Response Breakout	241
Reports Schemas Campaign Campaign Offer Response Breakout Columns [Response Type]	241
Reports Schemas Campaign Campaign Offer Contact Status Breakout	243
Reports Schemas Campaign Campaign Offer Contact Status Breakout Columns [Contact Status]	243
Reports Schemas Campaign Campaign Custom Attributes Columns [Campaign Custom Column]	244
Reports Schemas Campaign Campaign Custom Attributes Columns [Offer Custom Column]	245
Reports Schemas Campaign Campaign Custom Attributes Columns [Cell Custom Column]	246
Reports Schemas Interact	246
Reports Schemas Interact Interact Performance	247
Reports Schemas eMessage	248
IBM Marketing Operations configuration properties	248
Marketing Operations	248
Marketing Operations navigation	249
Marketing Operations about	250
Marketing Operations umoConfiguration	252
Marketing Operations umoConfiguration Approvals	256
Marketing Operations umoConfiguration templates	257
Marketing Operations umoConfiguration attachmentFolders	258
Marketing Operations umoConfiguration Email	260
Marketing Operations umoConfiguration markup	261
Marketing Operations umoConfiguration grid	262
Marketing Operations umoConfiguration workflow	264
Marketing Operations umoConfiguration integrationServices	265
Marketing Operations umoConfiguration campaignIntegration	266
Marketing Operations umoConfiguration reports	266
Marketing Operations umoConfiguration invoiceRollup	267
Marketing Operations umoConfiguration database	268

Marketing Operations umoConfiguration listingPages	270	Interact runtime environment configuration properties	391
Marketing Operations umoConfiguration objectCodeLocking	271	Interact design environment configuration properties	444
Marketing Operations umoConfiguration thumbnailGeneration.	273	Contact Optimization configuration properties	463
Marketing Operations umoConfiguration Scheduler intraDay.	274	Campaign unicaACOListener	463
Marketing Operations umoConfiguration Scheduler daily	274	Campaign unicaACOOptAdmin	466
Marketing Operations umoConfiguration Notifications.	274	Campaign partitions partition[n] Optimize sessionRunMonitor.	467
Marketing Operations umoConfiguration Notifications Email.	276	Campaign partitions partition[n] Optimize MemoryTuning	467
Marketing Operations umoConfiguration Notifications project	278	Campaign partitions partition[n] Optimize AgentTemplateTables	468
Marketing Operations umoConfiguration Notifications projectRequest.	280	Campaign partitions partition[n] Optimize userTemplateTables.	468
Marketing Operations umoConfiguration Notifications program	280	Campaign partitions partition[n] Optimize AlgorithmTuning	469
Marketing Operations umoConfiguration Notifications marketingObject	281	Campaign partitions partition[n] Optimize Debug	473
Marketing Operations umoConfiguration Notifications approval.	281	Campaign partitions partition[n] Optimize DatabaseTuning	473
Marketing Operations umoConfiguration Notifications asset	282	Campaign partitions partition[n] Optimize logging.	474
Marketing Operations umoConfiguration Notifications invoice	283	Distributed Marketing configuration properties	476
Campaign configuration properties	284	Collaborate navigation	477
Campaign	284	Collaborate UDM Configuration Settings	478
Campaign Collaborate.	285	IBM SPSS Modeler Advantage Marketing Edition configuration properties	504
Campaign navigation	285	SPSS integration.	504
Campaign caching	288	SPSS integration partitions partition [n]	505
Campaign partitions	289	SPSS navigation	505
Campaign monitoring	376	Opportunity Detection configuration properties	506
Campaign ProductReindex	378	Opportunity Detection	506
Campaign unicaACOListener	379	Opportunity Detection Navigation	507
Campaign server	384	Opportunity Detection System Interact	
Campaign logging	384	Design Service	508
eMessage configuration properties	384	Opportunity Detection System Interact	
eMessage serverComponentsAndLocations hostedServices	384	Connector	509
eMessage partitions partition[n] hostedAccountInfo	385	Opportunity Detection logging.	509
eMessage partitions partition[n] dataSources systemTables	386		
eMessage partitions partition[n] recipientListUploader.	389		
eMessage partitions partition[n] responseContactTracker	389		
Interact configuration properties	391		

Appendix B. Re-branding the frameset 511

To prepare your corporate theme	511
Applying your corporate theme	512

Before you contact IBM technical support 513

Notices 515

Trademarks	517
Privacy Policy and Terms of Use Considerations	517

Chapter 1. Introduction to the IBM Marketing Platform

The IBM® Marketing Platform provides security, configuration, and dashboard features for IBM EMM products.

- Support for reporting in for many products in IBM EMM.
- 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 EMM 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.
- A common user interface for IBM EMM products.

About IBM EMM security features

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

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

Security technologies used in IBM

Marketing Platform employs industry-standard encryption methods to perform authentication and enforce security across all IBM EMM 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 EMM 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

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 EMM 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. The IBM EMM 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

IBM EMM 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 Tivoli® Access Manager. This reduces errors, support costs, and the time needed to deploy an application in production.

Data filters

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

About configuration management

The Configuration page provides access to the central configuration properties for IBM EMM 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 a utility provided with Marketing Platform. See “configTool” on page 168 for details.

Localization in IBM EMM

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 local 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 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 EMM.

The common user interface

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

The common interface provides the following features.

- When multiple IBM EMM 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 EMM 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.

To log in to IBM EMM

Use this procedure to log in to IBM EMM.

Before you begin

You need the following.

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

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

Procedure

Access the IBM EMM URL using your browser.

- If IBM EMM 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 EMM 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 EMM displays the default dashboard page.

Results

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 administer IBM EMM dashboards, you must log in as `platform_admin`.

Chapter 2. User account management

This section describes how to 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.

You can manage internal accounts using the Marketing Platform user interface. External accounts are managed in the external system.

Types of user accounts: internal and external

When IBM EMM 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 EMM using the security user interface. These users are authenticated through IBM EMM.
- **External** – User accounts that are imported into IBM EMM through synchronization with an external server. This synchronization occurs only if IBM EMM 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 EMM with Windows Active Directory and enable Windows integrated login, you can have only external users.

For more information about integrating IBM EMM 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 EMM, you can control the following aspects of an external user account: data sources, notification preferences, locale preference for IBM EMM applications, and membership in internal groups (but not external groups).

Identifying internal and external users in the IBM EMM interface

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

- Internal - 
- External - 

Properties of user accounts

This section provides details on the properties of user accounts.

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 EMM 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 user is either active or disabled. A user with a disabled account cannot log in to any IBM EMM 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 EMM applications. You can enter these credentials in the user account properties.

When a user is working in an IBM EMM application such as Campaign and is prompted for data source information, the IBM EMM 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 EMM interface.

Adding a user account

Use this procedure to add a user account.

Procedure

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. See “**Users window reference**” on page 9 for allowed special characters.

4. Click **OK**.

The new user name appears in the list.

Deleting a user account

Use this procedure to delete a user account.

About this task

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.

Procedure

1. Click **Settings > Users**.
2. Click the user name of the account you want to delete.
3. Click the **Delete User** button above the account details in the right pane.
4. Click **OK**.

Changing an internal user password expiration date

If the system-wide password expiration property is set to never expire, you cannot change the password expiration date of an individual user.

Procedure

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 **PW expiration** field.
5. Click **OK**.

Resetting an internal user password

Use this procedure to reset an internal user password.

Procedure

1. Click **Settings > Users**.
The **Username** 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 EMM application.

Changing user account properties

Use this procedure to change user account properties.

Procedure

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.
To reset the password for internal accounts, see “Resetting an internal user password” on page 7.
5. Click **Save Changes** to save your changes.
6. Click **OK**.

Changing user system status

Use this procedure to change a user's system status.

Procedure

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 EMM applications. Users with Admin access to Marketing Platform cannot disable themselves.

5. Click **Save Changes** to save your changes.
6. Click **OK**.

Adding a user data source

Use this procedure to add a user data source.

Procedure

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**.

Changing a user data source password or login name

Use this procedure to change a data source password or login name.

Procedure

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. See “Changing user account properties” on page 7 for details.

- 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**.

Deleting a user data source

Use this procedure to delete a user data source.

Procedure

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**.

Users window reference

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

New User

Field	Description
First Name	The user's first name.
Last Name	The user's last name.
Login	<p>The user's login name. This is the only required field. Only the following special characters are allowed in login names.</p> <ul style="list-style-type: none">• Upper and lower case alphabetic characters (A–Za–z)• Numbers (0–9)• The ‘at’ sign (@)• Hyphen (-)• Underscore (_)• Dot (.)• Double byte characters (such as alpha-numeric Chinese characters) <p>Do not include other special characters (including spaces) in the login name.</p>
Password	<p>A password for the user. Follow these rules when creating a password.</p> <ul style="list-style-type: none">• 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 EMM. <p>Additional password requirements are set on the Configuration page. To see what they are for your installation of IBM EMM, click the Password Rules link next to the Password field.</p>
Confirm Password	The same password you entered in the Password field.
Title	The user's title.
Department	The user's department.

Field	Description
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 821 for details.
Alternate Login	The user's UNIX login name, if one exists. An alternate login is typically required when the Campaign listener 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 EMM applications.

Edit properties

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

Field	Description
Password	This field is not available on the Edit properties window.
Login	This field is not available on the Edit properties window.
PW 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 IBM Digital Analytics user name here as part of the configuration process. See Chapter 7, "Enabling single sign-on between IBM EMM and IBM Digital Analytics," on page 73 for complete details.

Reset password

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

New Data Source / Edit Data Source Properties

Field	Description
Data Source	The name of a data source you want the user to be able to access from an IBM EMM application. IBM EMM names 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). Required.

Field	Description
Data Source Login	The login name for this data source. Required.
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).

The per-user locale preference

For both internal and external users, you can set the locale preference on a per-user basis. This preference setting affects the display of language, time, numbers, and dates in IBM EMM applications.>

A default setting also exists that applies throughout IBM EMM for all users. See “About the default user locale preference” on page 31 for details.

When you set this property for an individual user, the setting you apply for that user overrides the default setting.

Note: Availability of locales may vary depending on the IBM EMM application, and not all IBM EMM applications support this locale setting. See specific product documentation to determine availability and support for the locale setting in IBM EMM.

Setting the user locale preference

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

Procedure

1. Click **Settings > Users**.
2. Click the user name you for which you want to set locale preferences.
3. Click the **Edit Configuration Preferences** link at the bottom of the page.
4. Click **Suite** in the left pane.
5. Select the option from the drop-down list.
6. Click **Save and Finish**.

Synchronization of external users

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

During these automatic synchronizations, only those users and groups that were created or changed since the last synchronization are brought into IBM EMM. You can force a synchronization of all users and groups by using the **Synchronize** function in the **Users** area of IBM EMM.

Use the procedure in this section to force synchronization of external users.

Forcing synchronization of external users

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

Procedure

1. Log in to IBM EMM 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 EMM applications.

For the most part, only Marketing Platform itself and Campaign use Marketing Platform's security functions to manage users' application access in detail. The other IBM EMM products use some basic application access roles set through Marketing Platform, and either do not have detailed security settings, or the settings are not managed in the Marketing Platform area of the user interface. IBM EMM products manage permissions as follows.

- For Marketing Platform, roles and permissions control users' access to Marketing Platform administration pages and their ability to modify user accounts other than their own account. You manage these roles on the User Roles & Permissions page.
Although the reporting function is a component of Marketing Platform, it has its own entry on the User Roles & Permissions page, and it has default roles with only broad, basic permissions.
- In Interaction History and Attribution Modeler, roles and permissions control users' access to their administration pages and their ability to view reports. You manage these roles on the User Roles & Permissions page.
- In Campaign, permissions control users' access to objects and their ability to perform various actions with objects. In Campaign only, permissions can apply to all objects within a folder, and multiple roles can be grouped into a policy, which can then be assigned to a user or group of users. You manage Campaign roles on the User Roles & Permissions page.
- For Marketing Operations, setting up the basic roles on the User Roles & 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 in the Marketing Operations area.
- CustomerInsight, Digital Analytics for On Premises, Distributed Marketing, eMessage, Interact, and Lead Referrals have default roles with broad, basic permissions for application access. They do not have permissions that allow you to define in detail a user's access to these applications.

Where to find information on security administration in IBM EMM

Information on security administration for IBM EMM is located in a several places.

- **All products that have roles and permissions in the Marketing Platform** - This guide provides information on assigning roles to users, either on a per-user basis or through group membership.
- **Marketing Platform, Interaction History, and Attribution Modeler** - This chapter provides the information you need to manage permissions for Marketing Platform, Interaction History, and Attribution Modeler.

For the reporting function, the basic permissions are described in this chapter, and additional details about how security works in reporting are described in the *IBM EMM Reports Installation and Configuration Guide*.

- **Campaign** - See the *IBM Campaign Administrator's Guide*.

- **CustomerInsight, Digital Analytics for On Premises, Distributed Marketing, eMessage, Interact, and Lead Referrals** - See “Reference: Permissions for products that use only basic roles” on page 25 for a description of the basic roles.
- **Marketing Operations** - See “Reference: Permissions for products that use only basic roles” on page 25 for a description of the basic roles. See the Marketing Operations product documentation for detailed information on setting up a security scheme.

About security administration 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 a user with the PlatformAdminRole role can administer users across partitions. Users with the AdminRole role can administer users in their own partition only.

Marketing Platform administrator uses the User Groups and User Roles & Permissions pages to perform the following tasks.

- 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 EMM 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

About 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)

About 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 EMM. 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.

The security management process 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.

To manage user application access

1. Plan the roles you want to use to control user access to Marketing Platform, Interaction History, Attribution Modeler, and Campaign. Configure these 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.

Types of groups: internal and external

When IBM EMM 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 EMM using the security user interface. These users are authenticated through IBM EMM.
- **External** – IBM EMM groups that are mapped to groups in the external system. This synchronization occurs only if IBM EMM has been configured to integrate with the external server. Examples of external servers are LDAP and web access control servers. Note that a group referred to as an external group in this guide is one that is actually created in IBM EMM 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 EMM 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 mapped 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.

About 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.

Pre-configured users and roles

When IBM EMM 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 EMM 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 EMM products beyond Marketing Platform and Campaign.

The asm_admin user account

The asm_admin user account is designed to allow an IBM EMM 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 EMM 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.

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 EMM 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 EMM. 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 as described in "restoreAccess" on page 177.

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.

Managing internal groups

This section describes how to manage internal groups.

Adding an internal group

Use this procedure to add an internal group.

Procedure

1. Click **Settings > User Groups**.
2. Click the **New Group** button above the **Group Hierarchy** list in the left pane.
3. Complete the **Group Name** and **Description** fields.

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.

Adding a subgroup

Use this procedure to add a subgroup.

Procedure

1. Click **Settings > User Groups**.
2. Click the name of the group to which you want to add a subgroup.
3. Click the **New Subgroup** button at the top of the right pane..
4. Complete the **Group Name** and **Description** fields.

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.

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

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.

Procedure

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**.

Changing a group or subgroup description

Use this procedure to change a group or subgroup description.

Procedure

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**.

Assigning a group to a partition

This procedure is necessary only if multiple partitions are configured for Campaign. Only the PlatformAdmin user can perform this task.

Procedure

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**.

Adding a user to a group or subgroup

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

Procedure

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.

Removing a user from a group or subgroup

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

About this task

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

Procedure

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**.

User Groups window reference

These are the fields you use to configure user groups.

New Group, New Subgroup, Edit Properties

Field	Description
Group Name	<p>The group name. The limit is 64 characters.</p> <p>You may use the following characters when you create a group name.</p> <ul style="list-style-type: none"> • 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) <p>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.</p> <p>IBM EMM names preserve case for display purposes, but use case-insensitive rules for comparison and creation (i.e., you cannot create both Admin and admin as separate group names).</p> <p>When you create a subgroup, it is a good idea to give your subgroup a name that relates it to its parent group.</p> <p>Required.</p>
Description	<p>The group description. The limit is 256 characters.</p> <p>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.</p>
Partition ID	<p>Available only when multiple partitions are configured.</p> <p>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.</p>

Edit Users, Edit Roles

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

Managing user roles and permissions

This section describes how to manage user application access through roles and permissions.

Creating a role

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

Procedure

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 and remain on the Properties/Roles page, or **Save and Edit Permissions** to go to the Permissions page to add or modify permissions for any of the roles in the list.

Modifying role permissions

Use this procedure to modify role permissions.

Procedure

1. Click **Settings > User Roles & Permissions**.
2. Click the plus sign next to **Campaign** or **Platform** 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 and return to the Properties/Roles page.

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.

Removing a role

Use this procedure to remove a role.

About this task

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

Procedure

1. Click **Settings > User Roles & Permissions**.
2. Click the plus sign next to **Campaign** or **Platform** 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**.

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.

Procedure

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**.

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.

Procedure

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 **Roles** box on the right.
4. Complete one of the following tasks:
Select a role in the **Available Roles** box.
 - 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 **Roles** box, and click **Remove**.
5. Click **Save Changes**, and then click **OK**.

Reference: Definition of permission states

For each role, you can specify which of the pre-defined permissions are granted, not granted, or denied.

These states have the following meanings.

- **Granted** — indicated with a green checkmark . Explicitly grants permission to perform this particular function as long as none of the user's other roles explicitly denies permission.
- **Denied** — indicated with a red "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 shaded gray "X" . 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.

Reference: Permissions for products that use only basic roles

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

IBM Application	Roles
Leads	Leads roles are reserved for future use.
Reports	<ul style="list-style-type: none">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 8 BI system only. <p>For information about authentication options for the IBM Cognos 8 BI integration and how the IBM Authentication Provider uses the reporting permissions, see the <i>IBM EMM Reports Installation and Configuration Guide</i>.</p>
eMessage	<ul style="list-style-type: none">eMessage_Admin – Has full access to all features.eMessage_User – Reserved for future use.
Interact	<ul style="list-style-type: none">InteractAdminRole – Has full access to all features.
Distributed Marketing	<ul style="list-style-type: none">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.

IBM Application	Roles
Marketing Operations	<ul style="list-style-type: none"> • 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. <p>Access is further defined through the security policies in Marketing Operations.</p>
IBM SPSS® Modeler Advantage Marketing Edition	<ul style="list-style-type: none"> • SPSSUser – Users with the SPSSUser role can do the following: <ul style="list-style-type: none"> – 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 Marketing Edition, including access to all administrative and configuration settings.

Reference: Permissions for Marketing Platform

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

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 EMM 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 Scheduled Tasks 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.

Reference: Permissions for Interaction History

The following table describes the permissions that you can assign to roles in Interaction History.

Permission	Description
Schedule ETL Jobs	Allows a user to schedule Interaction History ETL jobs and report generation on the Interaction History Settings page.
Define Mapping for Custom Columns	Allows a user to access the Campaign Audience Level Configurations window when you are configuring a Campaign ETL job on the Interaction History Settings page.
Define and Map Channels	Allows a user to access the Channel Mapping section on the Interaction History Settings page.
Define and Map Response Types	Allows a user to access the Response Type Mapping section on the Interaction History Settings page.
View Cross-Channel Reports	Allows a user to view reports that are part of the Interaction History reports pack.
View Admin Reports	Allows a user to view Interaction History administration reports.

Reference: Permissions for Attribution Modeler

The following table describes the permissions that you can assign to roles in Attribution Modeler.

Permission	Description
Schedule Attribution Modeler Jobs	Allows a user to schedule Attribution Modeler jobs.
Monitor Attribution Modeler Jobs	Allows a user to view the Scheduled Runs and Schedule Definitions pages, with only Attribution Modeler jobs that are listed.
Start/Pause/Resume/Stop Attribution Modeler Jobs	Allows a user to perform all actions on scheduled Attribution Modeler jobs.
View Reports	Reserved for future use.

Reference: Permissions for Opportunity Detection

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

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

Table 1. Permissions in Opportunity Detection

Permission	Description
View only	Access all of the user interface, in view-only mode.

Table 1. Permissions in Opportunity Detection (continued)

Permission	Description
Design triggers	<p>Create workspaces and design trigger systems in batch mode.</p> <p>Allows the following.</p> <ul style="list-style-type: none"> • Create, modify, and delete all trigger related resources. • Access Workspace, Component, Audience Level, Data Source, Named Value List pages. <p>Can not access Server Groups and Deployment Configuration, and cannot set off a batch run.</p>
Run for testing	<ul style="list-style-type: none"> • Deploy deployment configurations and run batch deployment configurations on server groups not designated for production. • Can access Server Group and Deployment Configuration pages, but can not designate a server group for production. <p>Can not access any production related resources.</p>
Run for production	<ul style="list-style-type: none"> • Deploy deployment configurations and run batch deployment configurations on any server group. • Perform all actions on the Server Groups, Deployment Configuration and Batch Run pages, including designating a server group for production.
Administer real time	<p>Manage objects that the web service creates when Opportunity Detection is integrated with Interact to enable real time mode.</p> <p>Allows the following.</p> <ul style="list-style-type: none"> • Delete workspaces and components created by the web service. • Start and stop real time deployment configurations and update their log level. <p>The user with this permission alone can not start runs for real time deployment configurations.</p> <p>No one, even with this permission, can do any of the following.</p> <ul style="list-style-type: none"> • 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.

Chapter 4. Configuration Management

When IBM EMM 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 EMM 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.

About property categories

The **Reports**, **General** and **Platform** categories are present when Marketing Platform is first installed. These categories contain the following properties that apply across all IBM EMM 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 are used to configure data filters
- Properties used to configure schedules
- Properties used to configure the reporting feature

Depending on the IBM EMM 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.

Identifying category types

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

Category type	Icon
Categories that contain no configurable properties	
Categories that contain configurable properties	
Template categories that you can use to create a category	

Duplicating categories using templates

The properties for an IBM EMM application are registered with Marketing Platform when the application is installed. When an application configuration requires that a category can be duplicated, 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.

Identifying category templates

The Configuration page shows category templates in the navigation tree. You can identify a category template in the tree because its label is in italics and enclosed in parentheses.

Naming a new category

The following restrictions apply when you name a new category.

- 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.

Deleting categories created from templates

By default, any category created from a template can be deleted.

Deleting categories

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 EMM product is registered, its set of categories might include categories that can be deleted.

Categories that can be deleted in the Configuration page have a **Delete Category** link on the Settings page. This page appears when you select the category in the navigation tree.

About property descriptions

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

- Click **Help > Help for this page** to launch online help. Click a product and then a configuration category in the pages that follow to navigate to the topic that describes all of the properties in a category.
- Click **Help > Product Documentation** to launch a page that gives you access to all of the product documentation in PDF format. All property descriptions are included as an appendix in the *IBM Marketing Platform Administrator's Guide*.

About refreshing the display

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 if 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.

About the default user locale preference

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

You can set this default by setting the value of the **Region setting** property in the **Suite** 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 EMM 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. See "The per-user locale preference" on page 11 for details.

Editing property values

This section describes how to edit property values on the Configuration page.

Navigating to a category

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

Procedure

1. Log in to IBM EMM.
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.

Procedure

1. Navigate to the category that contains the property you want to set, as described in “Navigating to a category” on page 31.
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 and Finish** to save your changes or **Cancel** to exit the page without saving.

Duplicating and deleting categories

This section describes how duplicate and delete categories on the Configuration page.

Creating a category from a template

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

Procedure

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 and Finish** to save the new configuration.

Results

The new category appears in the navigation tree.

Deleting a category

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

Procedure

1. On the Configuration page, navigate to the category you want to delete and click to select it.
The Settings page for the category appears.
2. Click the **Delete Category** link.
A window shows the message, Are you sure you want to delete "*category name*"?
3. Click **OK**.

Results

The category no longer appears in the navigation tree.

Chapter 5. Dashboards

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 EMM product pages, pages on your company intranet, or pages on the internet. See “Custom portlets” on page 50 for details on creating custom portlets.

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 EMM 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 EMM portlets.

User permissions required to view dashboards

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

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

- List: A list of IBM EMM 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 EMM report.

You can also create your own dashboard portlets, as described in “Custom portlet types and availability” on page 50.

Pre-defined portlet availability

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

You must enable the pre-defined portlets in Marketing Platform before you can use them in a dashboard. IBM EMM 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.

Pre-assembled dashboards

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

Pre-assembled dashboard availability

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 it is not available.

IBM EMM provides pre-defined dashboards. For a portlet included in a pre-assembled dashboard to be available, you must meet the following prerequisites.

- The product or products that support the portlet must be installed.
- The portlet must be enabled.

Reference: Pre-assembled dashboards

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

Pre-assembled dashboard	Purpose	Portlets	Required products
Campaign Management	This dashboard shows the financial results from campaigns.	<ul style="list-style-type: none"> • Financial Summary by Offer • Campaign Performance Comparison 	<ul style="list-style-type: none"> • Campaign • Campaign Report Pack
Project and Traffic Management	This dashboard provides status updates for projects.	<ul style="list-style-type: none"> • My Tasks • My Alerts • My Active Projects • My Task Summary • Projects Requested and Completed • Approvals Awaiting Action • My Approval Summary • Projects by Status 	<ul style="list-style-type: none"> • Marketing Operations • Marketing Operations Report Pack
Project Member	This dashboard shows tasks that require action and allows users to close completed tasks.	<ul style="list-style-type: none"> • My Tasks • My Active Projects • My Alerts • My Requests 	Marketing Operations
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.	<ul style="list-style-type: none"> • Approvals Awaiting Action • My Alerts • Marketing Financial Position • Projects by Project Type • Budget by Project Type • Spend by Project Type • Completed Projects by Quarter 	<ul style="list-style-type: none"> • 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.	<ul style="list-style-type: none"> • Approvals Awaiting Action • Marketing Financial Position • Alerts • Projects by Type • Completed Projects by Quarter 	<ul style="list-style-type: none"> • Marketing Operations with the Financial Management Module • Marketing Operations Report Pack

Pre-assembled dashboard	Purpose	Portlets	Required products
Cross-Channel Analytics	This dashboard shows marketing performance across multiple channels.	<ul style="list-style-type: none"> • Cross-Channel Summary • Channel-Campaign-Drilldown • Channel-Campaign-Offer Summary • Channel Attribution 	<ul style="list-style-type: none"> • Interaction History • Interaction History Report Pack • Attribution Modeler • Attribution Modeler 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 EMM 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 39.

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.

About this task

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

Procedure

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 EMM, add the report to the dashboard, if you have not done so already. See “Adding a pre-defined portlet to a dashboard” on page 47 or “Adding a user-created portlet to a dashboard” on page 53.
5. Only if the report is one of the pre-defined portlets, do the following in IBM EMM.
 - 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**.

Pre-defined portlet descriptions

This section provides descriptions of all of the IBM EMM pre-defined dashboard portlets, organized by product and portlet type.

Marketing OperationsIBM Cognos report portlets

This section describes the Marketing Operations dashboard portlets that are available after the Marketing Operations reports package is installed.

Table 2. Standard Marketing OperationsIBM Cognos report portlets

Report	Description
Budget by Project Type	An example IBM Cognos report showing 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 showing 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 showing a 3-D pie chart of the forecasted spending per project type for the current calendar year.

Table 2. Standard Marketing OperationsIBM Cognos report portlets (continued)

Report	Description
Manager Approval Summary	An example IBM Cognos report showing data for active and completed approvals for all In Progress projects in the system.
Manager Task Summary	An example IBM Cognos report showing data for active and completed tasks for all In Progress projects.
Marketing Financial Position	An example IBM Cognos report showing 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 showing data about all active and completed tasks for the user viewing the report in all In Progress projects.
My Approval Summary	An example IBM Cognos report showing data about active and completed approvals for the user viewing the report.
Projects by Project Type	An example IBM Cognos report showing 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 showing 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 showing 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 showing a 3-D pie chart of the actual amount spent per project type in the current calendar year. This report requires the Financial Management module.

Marketing Operations list portlets

This section describes the Marketing Operations list portlets that are available for use on dashboards even if the Marketing Operations reports package is not installed.

Table 3. Standard Marketing Operations list portlets

Report	Description
Approvals Awaiting Action	List of approvals waiting for your action.
Manage My Tasks	<p>Lists your Pending and Active tasks and Not Started and In Progress approvals. An option to change the status of each item appears.</p> <ul style="list-style-type: none"> • 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.

Table 3. Standard Marketing Operations list portlets (continued)

Report	Description
My Project Health	<p>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:</p> $\text{(Number of Finished Tasks + Number of Skipped Tasks) } \div \text{ Total Number of Workflow Tasks}$ <ul style="list-style-type: none"> To recalculate project health status, click . The system recalculates the health status for display by this portlet only, not for use elsewhere in Marketing Operations. <p>Note: Project health calculations can be made only at 5-minute intervals.</p> <ul style="list-style-type: none"> 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. To view the Summary tab for a project, click its name. 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. 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**.

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.
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.

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 "Configuring the Interaction Point Performance dashboard portlet."

Configuring 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 the version of Cognos you are using.

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

If you need 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.

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 4. 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.
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.

Attribution Modeler IBM Cognos report portlet

This section describes the dashboard portlets that are available in the Attribution Modeler reports package.

Report	Description
Channel-Campaign-Offer Attribution Drill-Down	An IBM Cognos report that shows the following data about each offer in each campaign in each channel. <ul style="list-style-type: none"> • Number of interactions • Number of responses • Percentage of responses • Revenue • Average revenue per response • Cost per response • Return on Investment (ROI)

Interaction History IBM Cognos report portlet

This section describes the dashboard portlets that are available in the Interaction History Cross-Channel reports package.

Report	Description
Cross-Channel Summary Over Time	An IBM Cognos report that shows the following data about each channel. <ul style="list-style-type: none"> • Number of campaigns • Number of interactions • Number of responses • Percentage of responses • Revenue • Average revenue per response • Cost per response • Return on Investment (ROI)

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 EMM 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.

See “About partitions and security management” on page 16 for information about how to set up partition membership for users.

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.

When you have multiple partitions configured, the procedure for setting up dashboards is as follows.

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.
See Chapter 4, "Managing Security in IBM EMM" for information about these tasks.
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 EMM.
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.
See “Enabling or disabling pre-defined portlets” on page 46 for details.
 - 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.

See the remainder of this chapter for detailed information about performing these tasks.

Enabling or disabling pre-defined portlets

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

Procedure

1. Log in to IBM EMM 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.

Creating a dashboard that is not pre-assembled

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

Procedure

1. In IBM EMM, select **Dashboard**
A Dashboard Administration page opens. All dashboards associated with your partition are shown.
2. Click **Create Dashboard**.
A **Create Dashboard** page opens.
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.

Creating a pre-assembled dashboard

Use this procedure to create a pre-assembled dashboard.

Procedure

1. Ensure that the portlets that comprise the pre-assembled dashboard you want to create are enabled.
See “Enabling or disabling pre-defined portlets” for details.
2. In IBM EMM, select **Dashboard**.
A Dashboard Administration page opens.
3. Click **Create Dashboard**.
A Create Dashboard page opens.
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.

Adding a pre-defined portlet to a dashboard

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

About this task

See “Custom portlet types and availability” on page 50 for information about adding user-created portlets to a dashboard.

Procedure

1. In IBM EMM, select **Dashboard** and then select the tab for the dashboard you want to work with.

2. Click **Manage Portlets**.

A Manage Portlets page opens, listing the 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.

Assigning or changing a dashboard administrator

Use this procedure to assign or change a dashboard administrator.

Procedure

1. In IBM EMM, 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**.

Removing a portlet from a dashboard

Use this procedure to remove a portlet from a dashboard.

Procedure

1. In IBM EMM, 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.

About this task

Procedure

1. In IBM EMM, 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.

Procedure

1. In IBM EMM, 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.

Procedure

1. In IBM EMM, 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.

Quick link portlets

Quick links are pre-defined links to IBM EMM products. Some quick links enable users to perform basic actions in the IBM EMM 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 EMM 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.

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.

Creating a quick link portlet

Use this procedure create a quick link portlet.

Procedure

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.
-

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 EMM pages.

- Any IBM EMM 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 EMM Reports Installation and Configuration Guide*.
- Quick links portlets, which you can build using pre-defined links to IBM EMM 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 EMM 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 IBM Digital Analytics. Also, if single-sign-on is enabled with IBM Digital Analytics, users can view IBM Digital Analytics reports in Marketing Platform dashboards without entering their credentials. Otherwise, users must enter their IBM Digital Analytics credentials to view IBM Digital Analytics reports in Marketing Platform dashboards.

Portlet creation process overview

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

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.
The following procedures describe how to prepare the URL for the various portlet sources.
 - Digital Analytics for On Premises on-premises report - "Preparing the URL from an on-premises Digital Analytics for On Premises report"
 - IBM Cognos report - "Preparing the URL from an IBM Cognos dashboard report" on page 52
 - IBM Digital Analytics report - "Preparing the URL from an IBM Digital Analytics report" on page 52
 - Digital Analytics for On Premises On Demand report and pages on the internet or your company intranet - "Preparing the URL from an intranet or internet page" on page 53
2. Add the portlet to a dashboard.
See "Adding a user-created portlet to a dashboard" on page 53.

Preparing the URL from an on-premises Digital Analytics for On Premises report

Use this procedure for reports in an on-premises Digital Analytics for On Premises installation.

Procedure

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. Encode the URL using a web tool that you can find by searching for "URL encoding" or "percent encoding."

7. Add the following to the beginning of the URL: *YourIBMEMMURL/suiteSignOn?target=* where *YourIBMEMMURL* is the login URL for your installation of IBM EMM.

For example, suppose you have the following information.

- Your IBM EMM URL is `http://myHost.myDomain:7001/unica`
- Your encoded Digital Analytics for On Premises report URL is `MyEncodedReportURL`

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

Preparing the URL from an IBM Cognos dashboard report

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

About this task

For information about creating dashboard reports with IBM Cognos, see the *IBM EMM 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 EMM 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 HTML-encoded name of the dashboard report. For example: `Campaign%20Performance%20Comparison`

For example,

`http://serverX.companyABC.com:7001/unica/reports/jsp/dashboard_portlet.jsp?product=Campaign&report=Campaign%20Performance%20Comparison`

If you have scheduled the report as described in "Scheduling a dashboard report" on page 39, add the following to the end of the URL:

`&isView=true`

Note: Encode the URL using a web tool that you can find by searching for "URL encoding" or "percent encoding."

Preparing the URL from an IBM Digital Analytics report

Use this procedure for IBM Digital Analytics reports.

About this task

If you want users to be able to view IBM Digital Analytics reports in dashboards without having to log in to IBM Digital Analytics, you must enable single sign-on between IBM EMM and IBM Digital Analytics. See Chapter 7, "Enabling single sign-on between IBM EMM and IBM Digital Analytics," on page 73 for details.

Procedure

1. Log in to IBM 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.

Preparing the URL from an intranet or internet page

Use this procedure for intranet or internet pages, including On-Demand Digital Analytics for On Premises pages.

Procedure

1. Point your browser to the desired page and copy the URL from your browser's address field.
2. Encode the URL using a web tool that you can find by searching for "URL encoding" or "percent encoding."

Adding a user-created portlet to a dashboard

Before performing this procedure, you should have prepared a URL as described elsewhere in this section.

Procedure

1. In IBM EMM, 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 IBM 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 IBM 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.

Manage Portlets window reference

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

Field	Description
Search by Source	Enter part or all of a product name to filter the portlet list based on the product that supplies the report.
Search by Portlet Name	Enter part or all of a portlet name to filter the portlet list based on the portlet names.
Reset	Click to clear the search filters.
Create Custom Portlet	Click to open a window where you can create a portlet that uses a URL you have obtained.
Create Quick Link Portlet	Click to open a window where you can create a quick link portlet.

Create Custom Portlet window reference

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

Field	Description
Type	Select the portlet type: a portlet that is not from IBM Digital Analytics, or a portlet that is from IBM 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 IBM Digital Analytics URL	Paste in your prepared URL.
Hidden Variables	Available only when the portlet is not from IBM 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.

Dashboard membership administration

Topics in this section describe how to manage dashboard membership.

About dashboard administration tasks

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.

Tasks related to modifying the layout and content of a dashboard are described in “Dashboard setup” on page 44.

Granting or removing dashboard membership

Use this procedure to grant or remove dashboard membership.

Procedure

1. In IBM EMM, 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**.

Chapter 6. Schedules

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

You can schedule process runs in several IBM EMM products, as follows.

- Campaign flowchart runs

Note: The IBM EMM Scheduler is completely independent of the Schedule process in Campaign. See “Difference between the IBM Campaign Schedule process and IBM EMM Scheduler” on page 64 for details.

- Contact Optimization optimization session and post-optimization flowchart runs
- eMessage mailings
- Attribution Modeler training and scoring runs
- Interaction History ETL jobs

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 EMM object, the start and end dates, and optionally, the frequency with which the task is run, (also called a recurrence pattern).
- A run is an execution instance of a schedule.

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

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.

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 EMM 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 179 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.

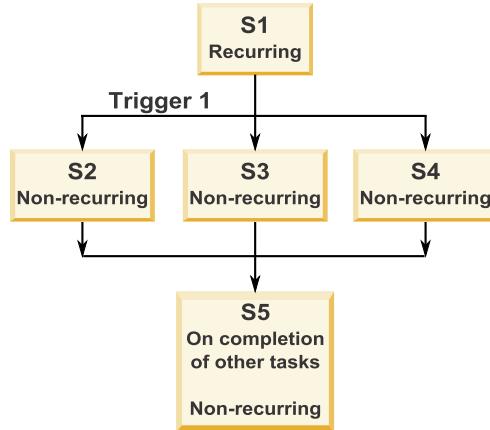
About start triggers

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, 2010 and on December 5, 2010 it receives its start trigger, the run will not start until December 12, 2010.

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 EMM 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. See “scheduler_console_client” on page 179 for complete details on using this utility.

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.

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 the Platform is installed. However, you can select from any other time zones listed in the **Select Timezone** 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 Scheduled Runs and Schedule Definitions pages
- 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 EMM 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.

Setting up throttling

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

Procedure

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 (throttling group) as described in “Creating a category from a template” on page 32.

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.

Best practices for setting up schedules

These are some best practices for planning and configuring scheduled runs of IBM EMM 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.

See “scheduler_console_client” on page 179 for complete details on using this utility.

Create or edit a schedule window reference

This section describes in detail the window you use when you create or edit a schedule.

Field	Description
Scheduled Item Type	The type of the scheduled object. This field is filled automatically, and is read-only.
Scheduled Item Name	The name of the scheduled object. This field is filled automatically, and is read-only.
Schedule Name	Enter a name for the schedule.
Description	Enter a description for the schedule.
Run Parameters	When you schedule a flowchart in Campaign, all of the values set on the Override Flowchart Parameters dialog are passed to the scheduler as a single string, displayed in the Run Parameters field. The run parameters are not used by the scheduler itself. The scheduler simply passes the string back to Campaign when the flowchart is run.
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. To appear as an option in this field, a group must be created using properties on the Configuration page.
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.
Select Timezone	Select the time zone to use when calculating the schedule, if you want a time zone that is different from the server time zone. See Time zone support for details.

Field	Description
When to start	<p>Select one of the following options to specify when 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 the schedule is configured to wait for a trigger, if it is a member of a throttling group, or if a recurrence pattern is in place.</p> <ul style="list-style-type: none"> • 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 one, you must configure a schedule to send this same string on success or failure. <p>Select one of the following options to specify the number of runs.</p> <ul style="list-style-type: none"> • Only run once - 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. See “Schedules that depend on completion of multiple runs” on page 58.
Recurrence Pattern	<p>Select one of the following options.</p> <ul style="list-style-type: none"> • 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.

What to consider when you use the scheduler with Campaign

Some special configuration applies when you use the IBM EMM 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 EMM Scheduler to run, and vice versa.

Difference between the IBM Campaign Schedule process and IBM EMM Scheduler

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

The IBM EMM 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 EMM 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 EMM 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 EMM Scheduler, it functions as configured; conditions required by the IBM EMM Scheduler and the Schedule process must be met before subsequent processes run.

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

Permissions for scheduling flowcharts

Scheduling Campaign flowcharts using the IBM EMM Scheduler requires the following permissions.

Table 5. 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.

Procedure

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

What to do next

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 EMM 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 with overrides of the default parameters

Follow this procedure to schedule a flowchart with parameter overrides.

Procedure

1. On a flowchart tab in **View** mode, click the **Run** icon and select **Schedule This - Advanced**.
The Override Flowchart Parameters dialog box opens.
2. Complete the fields in the dialog box to specify your flowchart parameters.
The system does not check syntax of the parameters you enter in this field.
Double-check that you have entered the correct values before proceeding.
3. Click **Schedule a Run**.
The Schedule flowchart dialog box appears.
4. 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.

5. Click **Run with this Schedule**.

What to do next

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.

Override Flowchart Parameters window reference

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.

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.
Schedule Job Name	Name for the scheduled job. This field defaults to the <i>CampaignName - FlowchartName</i> , but you can change the name to any name.
Catalog File Name	Specify a stored table catalog file to use for this run.
Data Sources	Use these fields to override the default login information for any of the data sources that this flowchart accesses.

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.

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.

Procedure

1. Select **Settings > Scheduled Tasks** and click the name of the schedule for which you want to set up notifications.
The Override Flowchart parameters window opens.
2. On the Override Flowchart Parameters window, click **Schedule a Run** to open the Edit a Schedule window.
3. Click **Edit Job Notifications** to open the All My Job Notifications window, and then click **New** to open the Add Job Notification window.
4. Complete the fields and click **Save**.

Deleting or modifying notifications for schedules you create

You can delete or modify any notifications you created.

Procedure

1. Select **Settings > My Job Notifications** to open the My Job Notifications 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.

Procedure

1. Create or identify one or more groups to use for schedule notification purposes.
For example, you might want to set up a group For each object type for which you want to enable group-based notifications.
2. Assign users to the group or groups as required.
3. For each object type for which you want to enable group-based notifications, set the value of the **Group Name to receive the Job Notifications** property to the name of the group you have set up to receive notifications for this object type.
4. Complete the fields and click **Save**.

Edit Job Notification window reference

You can use the information on the Edit Job Notification window to configure schedule notifications.

Table 6. Edit Job Notification

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 EMM user account, it can appear in your notifications in the user interface, or both.
Status	Select whether this notification is active or inactive. If you select inactive, no notifications are sent.

Schedule management pages

You can manage all schedules from pages you can access by selecting **Settings > Scheduled Tasks**. You must have the Scheduler Tasks View permission in Marketing Platform to have access to these pages.

These are the schedule management pages.

- Schedule Definitions - On this page you can view all schedule definitions or delete a definition. You can click the schedule name to edit a definition, including adding notifications. You can click the item name to go to that item in the relevant product.
- Scheduled Runs - On this page you can view queued and completed runs of every schedule, cancel a queued run, or delete a run. You can click the schedule name to edit a definition, including adding notifications. You can click the item name to go to that item in the relevant product.

In Campaign, to view the schedule management pages for a single flowchart, select **View when Scheduled** from the flowchart's **Run** menu.

Scheduled items in the list are links that take you directly to the flowchart.

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.

Scheduler management window reference

You access the scheduler management windows by selecting **Settings > Scheduled Tasks** or by selecting **View when Scheduled** from a flowchart's **Run** menu.

Scheduled Runs

Table 7. Fields on the Scheduled Runs page

Column	Description
Schedule Name	The schedule of which the run is an instance.
Scheduled Item	The name of the object to be run.
Item Type	The type of object to be run.
Start	Date and time when the first run of this task is scheduled.
Last Updated	The date and time of the most recent status update from the running flowchart or mailing process.
Run State	State of the run as defined in the scheduler, as follows. <ul style="list-style-type: none">• Scheduled - The run has not begun.• Queued - The scheduler has started the run, but the IBM EMM product has not begun executing the scheduled run due to throttling constraints.• Running - The run has started• 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, it is marked as cancelled, but this action does not stop the run.• Unknown - Indicates that Marketing Platform can not determine whether the job is still running, completed or failed.
Status	Status of the object's run as defined by the product. 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.

Schedule Definitions

Table 8. Fields on the Schedule Definitions page

Column	Definitions
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.
Created By	Login of the user who created the schedule.
Start Trigger	The string that, if received by this schedule, initiates a run. This field is blank if no start trigger is specified.
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.

Table 8. Fields on the Schedule Definitions page (continued)

Column	Definitions
Previous 1 & Next 2 Runs	Date and time of the previous run and next two scheduled runs. Note: Applies to recurring scheduled tasks only.
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.

Chapter 7. Enabling single sign-on between IBM EMM and IBM Digital Analytics

If your organization uses IBM Digital Analytics, you can enable single sign-on between IBM Digital Analytics and IBM EMM.

Single sign-on allows users to navigate to IBM Digital Analytics reports from within the IBM EMM user interface without being prompted to log in.

Also, if IBM Digital Analytics reports are referenced in IBM EMM dashboards, single sign-on allows users to view these reports (if they have access to them in IBM Digital Analytics).

Two options for enabling single sign-on between IBM EMM and IBM Digital Analytics

You can choose between two options for enabling single sign-on.

- You can configure IBM Digital Analytics to automatically create an IBM Digital Analytics user account the first time an IBM EMM user navigates to IBM Digital Analytics.

You might want to choose this option if you want all of your IBM EMM users to have single sign-on with IBM Digital Analytics.

See “Setting up single sign-on between IBM EMM and IBM Digital Analytics using automatic user account creation” on page 74.

- You can configure IBM EMM user accounts for single sign-on by adding each user's existing IBM Digital Analytics login name to his or her detail page in IBM EMM.

When you choose this option, users who require access to IBM Digital Analytics must have an IBM Digital Analytics account.

You might want to choose this option if you want a subset of your IBM EMM users to have single sign-on with IBM Digital Analytics.

See “Setting up single sign-on between IBM EMM and IBM Digital Analytics using manual user account creation” on page 75 for details.

Permissions in IBM Digital Analytics for single sign-on users

When the automatic account creation option is **not** selected in IBM Digital Analytics, single sign-on users have the permissions in IBM Digital Analytics that they would have if they log in to IBM Digital Analytics directly.

When the automatic account creation option is selected in IBM Digital Analytics, single sign-on users have permissions in IBM Digital Analytics as follows.

- By default, users have the permissions granted to the IBM 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 IBM 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 IBM Digital Analytics directly.

See “Setting up single sign-on between IBM EMM and IBM Digital Analytics using automatic user account creation” for details.

Server clock coordination

The clock on the server on which Marketing Platform is deployed must match the time on the IBM Digital Analytics server clock. For single sign-on, the IBM 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 EMM and IBM Digital Analytics using automatic user account creation

Use this procedure to set up single sign-on between IBM EMM and IBM Digital Analytics using automatic user account creation.

Procedure

1. Determine the IBM Digital Analytics Client ID you want to use for single sign-on between IBM EMM and IBM Digital Analytics.

Make a note of the Client ID, as you will need it in a later step.
 2. Log in to IBM 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 EMM 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.

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 IBM 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 IBM Digital Analytics account.

Note: If an account does not exist in IBM 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. Perform the procedure described in “Configuring your web application server for single sign-on between IBM Digital Analytics and IBM EMM” on page 77.

Setting up single sign-on between IBM EMM and IBM Digital Analytics using manual user account creation

Use this procedure to set up single sign-on between IBM EMM and IBM Digital Analytics using manual user account creation.

Procedure

1. Determine the IBM Digital Analytics Client ID you want to use for single sign-on between IBM EMM and IBM Digital Analytics.

Make a note of the Client ID, as you will need it in a later step.

2. Log in to IBM 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 EMM 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.

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 IBM Digital Analytics login name in the **Digital Analytics Username** field on the user's Edit Properties page.

Note: If a user has exactly the same login names in both IBM EMM and IBM Digital Analytics, you do not have to perform this step.

9. Perform the procedure described in “Configuring your web application server for single sign-on between IBM Digital Analytics and IBM EMM.”

Configuring your web application server for single sign-on between IBM Digital Analytics and IBM EMM

Perform the appropriate procedure below in the web application server where Marketing Platform is deployed to ensure that users can view IBM Digital Analytics reports in dashboards without having to log in.

WebLogic configuration for single sign-on

Edit the `setDomainEnv` script, located in the `bin` directory under your WebLogic domain directory, as follows.

Add the following to `JAVA_OPTIONS`.

```
-Dweblogic.security.SSL.ignoreHostnameVerification=true
```

WebSphere® configuration for single sign-on

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**.
7. Complete fields as follows.
 - **Host name:** welcome.coremetrics.com
 - **Port:** 443
 - **Alias:** coremetrics_cert

Chapter 8. Integration between IBM EMM 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 EMM 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 EMM 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 EMM applications query Marketing Platform for user authorization information. When Active Directory server integration is implemented and Windows integrated login is enabled, users are authenticated to all IBM EMM applications when they log in to the corporate network, and no password is required to log in to IBM EMM applications. User authentication is based on their Windows login, bypassing the applications' login screens.

If Windows integrated login is not enabled, users must still log in on the IBM EMM login screen, using their Windows credentials.

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).

Managing internal and external users

When Windows integrated login is enabled, all users are created and maintained in the Active Directory server. (You do not have the option of creating some users in Marketing Platform, which are known as internal users in this guide). If you require the ability to create internal users, do not enable Windows integrated login.

If you prefer not to enable Windows integrated login, follow the directions for integrating with an LDAP server. See “Configuration process roadmap (LDAP integration)” on page 93 for details.

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.

Importing users based on groups or attributes

You can choose one of three types of filtering to select the user accounts that are imported from the LDAP server into Marketing Platform.

You must choose between group based, attribute based, or Distinguished Name 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 maintained.

You can assign IBM EMM privileges by mapping an Active Directory group to an IBM EMM group. This mapping allows any new users added to the mapped Active Directory group to assume the privileges set for the corresponding IBM EMM 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 EMM 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 attributes on which you want to filter.
2. Set the **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 EMM 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 EMM 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.

Synchronization

When IBM EMM is configured to integrate with an Active Directory server, users and groups are synchronized automatically at pre-defined intervals. During these automatic synchronizations, only those users and groups (specified by the configuration) that were created or changed since the last synchronization are brought into IBM EMM. You can force a synchronization of all users and groups by using the Synchronize function in the Users area of IBM EMM.

You can also force a full synchronization, as described in “Forcing synchronization of external users” on page 11.

Special characters in login names

LDAP users with special characters in their login names may experience problems with authentication. See “Users window reference” on page 9 for a list of allowed special characters. For LDAP accounts that you plan to import into IBM EMM, change login names that contain special characters that are not allowed.

Active Directory integration prerequisites

To take advantage of the Windows Active Directory integration features, IBM EMM applications must be installed on a supported operating system.

In addition, to implement Windows integrated login, users accessing IBM EMM 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 EMM is authenticating.
- Use a supported browser. If Windows integrated login is enabled, the browser must support NTLM authorization.

Configuration process roadmap (Active Directory integration)

Use this configuration process roadmap to scan the tasks required to integrate IBM EMM with Windows Active Directory. The Topic column provides links to the topics that describe the tasks in detail.

Topic	Information
“Required information”	Obtain information about your Windows Active Directory server, which is required for integration with IBM EMM.
“Group membership, mapping, and application access” on page 84	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 84	If your directory server does not allow anonymous access (the most common configuration), configure an IBM EMM user account to hold a directory server administrator user name and password.
• “Setting connection properties” on page 85 • “Setting LDAP synchronization properties” on page 85 • “Setting user attributes map properties” on page 86 • “Mapping LDAP groups to IBM EMM groups” on page 86	Configure the Marketing Platform for integration by setting values on the Configuration page.
“Testing synchronization” on page 87	Verify that users are imported as expected, and if you are using group based synchronization, verify that users and groups are synchronizing properly.
“Setting up an Active Directory user with PlatformAdminRole permissions” on page 87	Set up administrator access to Marketing Platform, required when Windows integrated login is enabled.
“Setting the security mode to Windows Integrated Login” on page 88	Set the security mode values on the Configuration page.
“Configuring Internet Explorer” on page 88	Set a custom security level in every instance of Internet Explorer that is used to access IBM EMM. This is required with Windows integrated login, to prevent users from being presented with the IBM EMM login screen.
“Restarting the web application server” on page 88	This step is required to ensure that all of your changes are applied.
“Testing login as an Active Directory user” on page 88	Verify that you can log in to IBM EMM as an Active Directory user.

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 set configuration property values.

- 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). For additional information , see “About Distinguished Names” on page 83.
- 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.
 - Determine the string that your directory server uses for the alternate login attribute, which is required only when Campaign is installed in a UNIX environment.
- 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 the 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

About Distinguished Names

To enable directory server integration in IBM EMM, 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 EMM 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 EMM user account to hold the user name and password of a directory server user, as described in the following procedure.

Procedure

- Log in to IBM EMM as a user with Admin access.
- Select or create an IBM EMM 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 EMM 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 EMM user account must not match the user name of any directory server user.
- Add a data source for this IBM EMM user account to store the credentials that Marketing Platform uses to connect with the LDAP server. Follow these guidelines.

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 <code>LDAPServer</code> .
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 EMM. The DN resembles the following: <code>uidcn=user1,ou=someGroup,dc=systemName,dc=com</code>
Data Source Password	Enter the password of the administrative user with search permission on the directory server.

Setting connection properties

LDAP login method properties specify connection details the system uses to connect to the directory server.

Procedure

1. Click **Settings > Configuration** and navigate to the **IBM EMM | Platform | Security | Login method details | LDAP** category.
2. Set values of the following configuration properties.

See each property's context help, or the PDF version of the *IBM Marketing Platform Administrator's Guide*, for information about how to set the values.

- 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

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.

Procedure

1. Click **Settings > Configuration** and navigate to the **IBM EMM | 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 PDF version of the *IBM Marketing Platform Administrator's Guide*, for information about how to set 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)

Setting user attributes map properties

These properties specify the user attributes that the system imports from the directory server.

Procedure

1. Click **Settings > Configuration** and navigate to the **IBM EMM | Platform | Security | LDAP Synchronization** category.
2. Set values in the **User attributes map** section to map the listed IBM EMM 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 in the earlier step, "Obtain 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 the Marketing Platform synchronizes with your directory server.
- The 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.

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

Mapping LDAP groups to IBM EMM 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.

About this task

Important: Do not map any of the groups that have the `asm_admin` user as a member.

Procedure

1. Click **Settings > Configuration** and navigate to the **IBM EMM | 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 IBMEMMPlatformUsers` group to the Marketing Platform `marketingopsUsers` and `campaignUsers` groups (FILTER is omitted).

- LDAP reference: `cn=IBMEMMPlatformUsers,cn=Users, dc=myCompany,dc=com`
- IBM Marketing Platform group: `marketingopsUsers;campaignUsers`

Testing synchronization

Verify that users and groups are correctly synchronized between your servers.

Procedure

1. Log in to IBM EMM as an IBM EMM 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 Windows integrated login is enabled, you can not log in to IBM EMM as `platform_admin`, so you must perform the following procedure in order to have administrator access to the Marketing Platform.

Procedure

1. Log in to IBM EMM as an internal user (a user created in the 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.

Setting the security mode to Windows Integrated Login

Set security mode properties to allow Active Directory users to access IBM EMM applications based on their Windows login, bypassing the IBM EMM login screen.

Procedure

1. Click **Settings > Configuration** and, navigate to **IBM EMM | Platform | Security**.
2. Set the value of the **Login method** property to **Windows Integrated Login**.
3. Navigate to **IBM EMM | Platform | Security | Login method details | Windows integrated login** and set the values of the following properties.
 - Domain
 - Client Timeout
 - Cache Policy
 - Domain Controller
 - IP of the WINS server
 - Strip Domain
 - Retry on Authentication Failure

Restarting the web application server

Restart the web application server to ensure that all of your configuration changes are applied.

Configuring Internet Explorer

Perform this task in every instance of Internet Explorer that is used to access IBM EMM. This is required with Windows integrated login, to prevent users from being presented with the IBM EMM login screen.

Procedure

1. Open Internet Explorer and select **Tools > Internet Options**.
2. On the **Security** tab, click **Custom Level**.
3. In the **User Authentication** section, select **Automatic logon with current user name and password**.

What to do next

See the following links for additional information that can help you to troubleshoot browser-related login problems with Windows integrated login.

- <http://support.microsoft.com/default.aspx?scid=kb;en-us;Q258063>
- <http://support.microsoft.com/default.aspx?scid=kb;en-us;Q174360>
- <http://support.microsoft.com/default.aspx?scid=kb;en-us;Q303650>

Testing login as an Active Directory user

Verify the configuration by logging in to IBM EMM with an appropriate Windows Active Directory user account.

Procedure

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 EMM URL.

You should not see the IBM EMM login screen, and you should be allowed to access the IBM EMM user interface. If you cannot log in, see “restoreAccess” on page 177.

Chapter 9. Integration between IBM EMM 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 EMM 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 EMM 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 EMM applications query the Marketing Platform for user authorization information. When LDAP integration is implemented, users enter their valid LDAP user name and password for authentication to IBM EMM applications.

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 the 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).

Managing internal and external users

When integration is configured, you cannot add, modify, or delete the imported user accounts in the 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 the Marketing Platform, users may encounter problems with authentication.

Any user accounts you delete on the LDAP side are not deleted from the Marketing Platform. You should disable these accounts manually in the 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.

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 the Marketing Platform.

You must choose between group based or attribute based import; multiple methods are not supported simultaneously.

Group based import

The 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 the Marketing Platform imports users and groups from the server database, group memberships are maintained.

Note: The LDAP groups must have a unique name even if the groups are configured for separate partitions.

You can assign IBM EMM privileges by mapping an LDAP group to an IBM EMM group. This mapping allows any new users added to the mapped LDAP group to assume the privileges set for the corresponding IBM EMM group.

A subgroup in the 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 EMM 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 attributes on which you want to filter.
2. Set the **Platform | Security | LDAP synchronization | LDAP user reference attribute name** property to DN.

This indicates to the 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 EMM 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 EMM 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 EMM supports two types of user accounts and groups.

- **Internal** – User accounts and groups that are created within IBM EMM using the IBM EMM security user interface. These users are authenticated through Marketing Platform.
- **External** – User accounts and groups that are imported into IBM EMM through synchronization with a supported LDAP server. This synchronization occurs only if IBM EMM 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 EMM 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.

Synchronization

When IBM EMM is configured to integrate with an LDAP server, users and groups are synchronized automatically at pre-defined intervals.

During these automatic synchronizations, only those users and groups (specified by the configuration) that were created or changed since the last synchronization are brought into IBM EMM. You can force a synchronization of all users and groups by using the Synchronize function in the Users area of IBM EMM.

You can also force a full synchronization, as described in “Forcing synchronization of external users” on page 11.

Special characters in login names

LDAP users with special characters in their login names may experience problems with authentication. See “Users window reference” on page 9 for a list of allowed special characters. For LDAP accounts that you plan to import into IBM EMM, change login names that contain special characters that are not allowed.

LDAP integration prerequisites

To take advantage of the LDAP integration features, IBM EMM 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 EMM with LDAP. The Topic column provides links to the topics that describe the tasks in detail.

Topic	Information
“Required information” on page 82	Obtain information about your LDAP server, which is needed for integration with IBM EMM.

Topic	Information
“Group membership, mapping, and application access” on page 84	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 84	If your directory server does not allow anonymous access (the most common configuration), configure an IBM EMM user account to hold a directory server administrator user name and password.
<ul style="list-style-type: none"> • “Setting connection properties” on page 85 • “Setting LDAP synchronization properties” on page 85 • “Setting user attributes map properties” on page 86 • “Mapping LDAP groups to IBM EMM groups” on page 86 	Configure the Marketing Platform for integration by setting values on the Configuration page.
“Testing synchronization” on page 87	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 99	Set the security mode values on the Configuration page.
“Restarting the web application server” on page 88	This step is required to ensure that all of your changes are applied.
“Testing login as an LDAP user” on page 99	Verify that you can log in to IBM EMM as an LDAP user.

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 set configuration property values.

- 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). For additional information , see “About Distinguished Names” on page 83.
- 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.
 - Determine the string that your directory server uses for the alternate login attribute, which is required only when Campaign is installed in a UNIX environment.
- 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 the 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

About Distinguished Names

To enable directory server integration in IBM EMM, 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 EMM 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 EMM user account to hold the user name and password of a directory server user, as described in the following procedure.

Procedure

1. Log in to IBM EMM as a user with Admin access.
2. Select or create an IBM EMM 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 EMM 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 EMM user account must not match the user name of any directory server user.
3. Add a data source for this IBM EMM user account to store the credentials that Marketing Platform uses to connect with the LDAP server. Follow these guidelines.

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 <code>LDAPServer</code> .
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 EMM. The DN resembles the following: <code>uidcn=user1,ou=someGroup,dc=systemName,dc=com</code>
Data Source Password	Enter the password of the administrative user with search permission on the directory server.

Setting connection properties

LDAP login method properties specify connection details the system uses to connect to the directory server.

Procedure

1. Click **Settings > Configuration** and navigate to the **IBM EMM | Platform | Security | Login method details | LDAP** category.
2. Set values of the following configuration properties.

See each property's context help, or the PDF version of the *IBM Marketing Platform Administrator's Guide*, for information about how to set the values.

- 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

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.

Procedure

1. Click **Settings > Configuration** and navigate to the **IBM EMM | 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 PDF version of the *IBM Marketing Platform Administrator's Guide*, for information about how to set 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)

Setting user attributes map properties

These properties specify the user attributes that the system imports from the directory server.

Procedure

1. Click **Settings > Configuration** and navigate to the **IBM EMM | Platform | Security | LDAP Synchronization** category.
2. Set values in the **User attributes map** section to map the listed IBM EMM 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 in the earlier step, "Obtain 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 the Marketing Platform synchronizes with your directory server.
- The 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.

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

Mapping LDAP groups to IBM EMM 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.

About this task

Important: Do not map any of the groups that have the `asm_admin` user as a member.

Procedure

- Click **Settings > Configuration** and navigate to the **IBM EMM | Platform | Security | LDAP Synchronization | LDAP reference to IBM Marketing Platform group map** category.
- 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 IBMEMMPlatformUsers` group to the Marketing Platform `marketingopsUsers` and `campaignUsers` groups (`FILTER` is omitted).

- LDAP reference: cn=IBMEMMP1platformUsers,cn=Users, dc=myCompany,dc=com
- IBM Marketing Platform group: marketingopsUsers;campaignUsers

Testing synchronization

Verify that users and groups are correctly synchronized between your servers.

Procedure

1. Log in to IBM EMM as an IBM EMM 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 EMM applications.

Procedure

1. Log in to IBM EMM, click **Settings > Configuration**, and navigate to **IBM EMM | 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 EMM 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.

About this task

Chapter 10. 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 EMM 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 EMM are authenticated to all IBM EMM applications when they log in to the web access control system. These users do not see the IBM EMM application login screens.

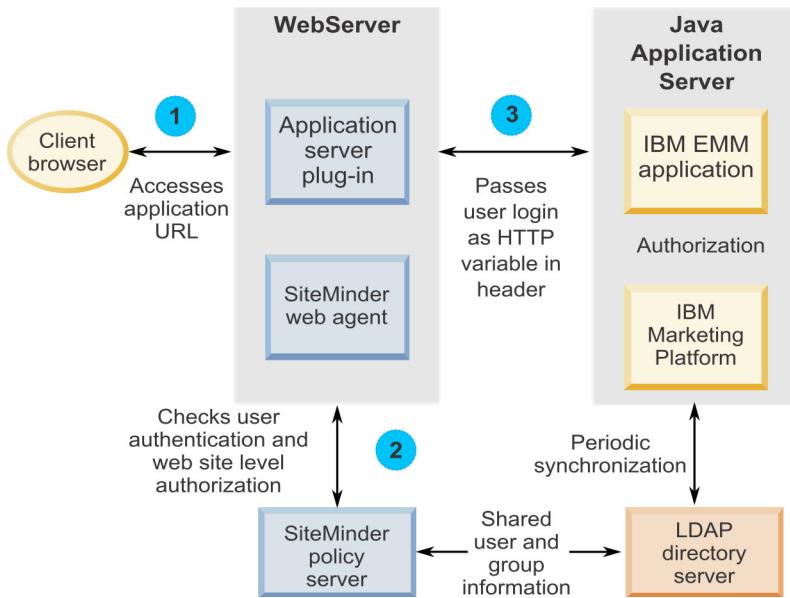
Authorization

IBM EMM 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 EMM are referencing a consistent set of users.

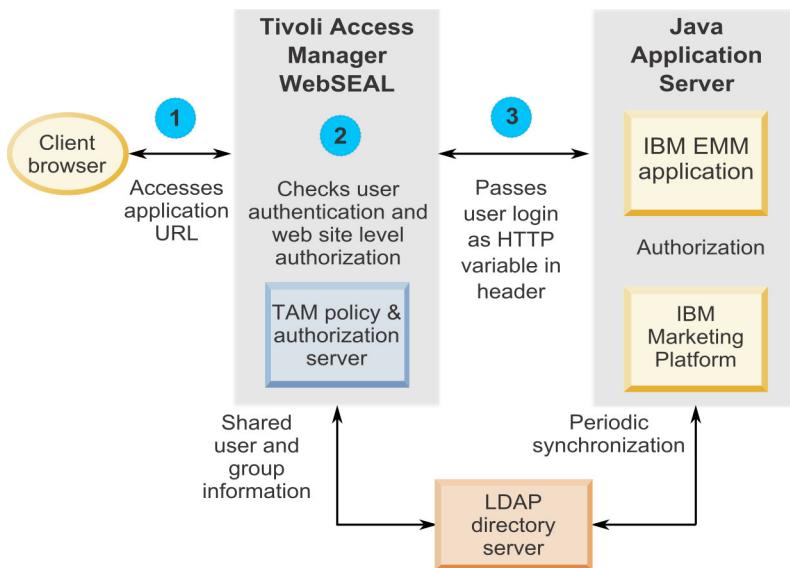
Additional authorization controls, including control over the application URLs to which users have access, are also available through most web access control systems.

Web access control integration diagrams

The following figure illustrates how IBM EMM works with SiteMinder and an LDAP directory server to authenticate and authorize users.



The following figure illustrates how IBM EMM works with Tivoli Access Manager and an LDAP directory server to authenticate and authorize users.



SiteMinder integration prerequisites

The following prerequisites must be met to integrate IBM EMM 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 EMM application, and the IBM EMMWeb access control header `variable` property must be set to the name of this variable (by default, `sm_user`).
- The SiteMinder policy server must be configured to use LDAP as its repository for storing group members, and user properties.

- The IBM EMM application URLs provided by the web server hosting SiteMinder and the Java™ application server hosting the IBM EMM application must refer to the same path.
- The web server hosting SiteMinder must be configured to redirect requests to the IBM EMM application URL on the Java application server.
- All users who need to access IBM EMM applications must be granted access in SiteMinder to the IBM EMM 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 products.

Enabling single logouts

To enable a logout of SiteMinder when a user logs out of an IBM EMM application, configure SiteMinder as follows.

1. Log in to the **Administer Policy Server** area of SiteMinder and set the `logoffUri` property to the URL of the IBM EMM 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 EMM logout page, to ensure that SiteMinder does not force the user to sign in again to view the logout page.

Enabling the IBM EMM Scheduler

If you plan to use the IBM EMM Scheduler, you must configure SiteMinder as follows.

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. In the **Resource Filter** text box, enter `/unica/servlet/SchedulerAPIServlet`.
4. Under **Default Resource Protection**, select **Unprotected**.

Enabling the IBM EMM Data Filter

If you plan to use the IBM EMM Data Filter, you must configure SiteMinder as follows.

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. In the **Resource Filter** text box, enter `/unica/servlet/DataFiltering`.
4. Under **Default Resource Protection**, select **Unprotected**.

Configuring settings for IBM Campaign

If you plan to use IBM Campaign, you must configure SiteMinder as follows.

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. In the **Resource Filter** text box, enter
`/Campaign/services/CampaignServices30Service`.
4. Under **Default Resource Protection**, select **Unprotected**.

5. Repeat the previous two steps, entering the following string in the **Resource Filter** text box.
`/Campaign/FlowchartNotifyScheduler`

Configuring settings for IBM Contact Optimization

If you plan to schedule IBM Contact Optimization sessions, you must configure SiteMinder as follows.

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. In the **Resource Filter** text box, enter
`/Campaign/optimize/ext_runOptimizeSession.do`.
4. Under **Default Resource Protection**, select **Unprotected**.
5. Repeat the previous two steps, entering the following strings in the **Resource Filter** text box.
 - `/Campaign/optimize/ext_optimizeSessionProgress.do`
 - `/Campaign/optimize/ext_doLogout.do`

Configuring settings for IBM Marketing Operations

If you plan to use Marketing Operations, you must configure SiteMinder as follows.

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. In the **Resource Filter** text box, enter
`/plan/errorPage.jsp`.
4. Under **Default Resource Protection**, select **Unprotected**.
5. Repeat the previous two steps, entering the following strings in the **Resource Filter** text box.
 - `/plan/alertsService`
 - `/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`

Configuring settings for IBM Distributed Marketing

If you plan to use Distributed Marketing, you must configure SiteMinder as follows.

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. In the **Resource Filter** text box, enter
`/collaborate/affiniumcollaborate.jsp`.

4. Under **Default Resource Protection**, select **Unprotected**.
5. Repeat the previous two steps, entering the following strings in the **Resource Filter** text box.
 - /collaborate/services/CollaborateIntegrationServices1.0
 - /collaborate/flowchartRunNotifyServlet
 - /collaborate/js/js_messages.jsp
 - /collaborate/js/format_symbols.jsp
 - /collaborate/alertsService

Configuring settings for IBM Interact

If you plan to use Interact, you must configure SiteMinder as follows.

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. In the **Resource Filter** text box, enter
`/Campaign/interact/flowchartEventPatterns.udo`.
4. Under **Default Resource Protection**, select **Unprotected**.
5. Repeat the previous two steps, entering the following strings in the **Resource Filter** text box.
 - /Campaign/interact/saveFlowchartAction.udo
 - /Campaign/interact/testRunFlowchart.udo
 - /Campaign/interact/getProfileDataAction.udo
 - /Campaign/interact/manageIPB.udo
 - /Campaign/initOfferListResolution.udo
 - /Campaign/getOfferListResolutionStatus.udo

Configuring settings for IBM SPSS Modeler Advantage Marketing Edition

If you plan to use IBM SPSS Modeler Advantage Marketing Edition, you must configure SiteMinder as follows.

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. In the **Resource Filter** text box, enter
`/unica/rest/spssUser`.
4. Under **Default Resource Protection**, select **Unprotected**.

Tivoli Access Manager integration prerequisites

The following prerequisites must be met to integrate IBM EMM with IBM Tivoli Access Manager.

- The Tivoli 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 EMM application, and the IBM EMM Web access control header variable property must be set to the name of this user name variable (by default, `iv-user`).
- The Tivoli Access Manager policy server must be configured to use LDAP as its repository for storing group members and user attributes.

- The IBM EMM application URLs defined by a WebSEAL junction and the Java application server hosting the IBM EMM application must refer to the same path.
- All users who need to access IBM EMM 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 EMM application, they are not automatically logged out of Tivoli Access Manager. They must close their browser after they log out of an IBM EMM application to log out of Tivoli Access Manager.

Enabling the IBM EMM Scheduler

If you plan to use the IBM EMM Scheduler, you must configure an Access Control List (ACL) policy in Tivoli as follows.

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 an un-protected object. Use the complete scheduler servlet path in Tivoli, starting from WebSEAL. The Marketing Platform context root is typically **unica**.
WebSEAL junction/Marketing Platform context root/servlet/SchedulerAPIServlet.

Enabling the IBM EMM Data Filter

If you plan to use the IBM EMM Data Filter, you must configure an Access Control List (ACL) policy in Tivoli as follows.

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 an un-protected object. Use the complete data filter servlet path in Tivoli, starting from WebSEAL. The Marketing Platform context root is typically **unica**.
WebSEAL junction/Marketing Platform context root/servlet/DataFiltering.

Configuring settings for IBM Campaign

If you plan to schedule IBM Campaign, you must configure an Access Control List (ACL) policy in Tivoli as follows.

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 the following as un-protected objects. Use the complete path in Tivoli, starting from WebSEAL. The Campaign context root is typically Campaign.
 - *WebSEAL junction/Campaign context root/services/ CampaignServices30Service*
 - *WebSEAL junction/Campaign context root/FlowchartNotifyScheduler*

Configuring settings for IBM Contact Optimization

If you plan to schedule IBM Contact Optimization sessions, you must configure an Access Control List (ACL) policy in Tivoli as follows.

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 the following as un-protected objects. Use the complete path in Tivoli, starting from WebSEAL. The context root is typically Campaign/optimize.
 - *WebSEAL junction/context root/ext_runOptimizeSession.do*
 - *WebSEAL junction/context root/ext_optimizeSessionProgress.do*
 - *WebSEAL junction/context root/ext_dologout.do*

Configuring settings for IBM Marketing Operations

If you plan to schedule IBM Marketing Operations, you must configure an Access Control List (ACL) policy in Tivoli as follows.

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 the following as un-protected objects. Use the complete path in Tivoli, starting from WebSEAL. The Marketing Operations context root is typically `plan`, and Marketing Platform context root is typically `unica`.
 - `WebSEAL junction/Marketing Operations context root/errorPage.jsp`
 - `WebSEAL junction/Marketing Operations context root/alertsService`
 - `WebSEAL junction/Marketing Operations context root/services/collabService`
 - `WebSEAL junction/Marketing Operations context root/services/PlanIntegrationServices/1.0`
 - `WebSEAL junction/Marketing Operations context root/affiniumplan.jsp`
 - `WebSEAL junction/Marketing Operations context root/invalid_user.jsp`
 - `WebSEAL junction/Marketing Operations context root/js/js_messages.jsp`
 - `WebSEAL junction/Marketing Operations context root/js/format_symbols.jsp`
 - `WebSEAL junction/Marketing Platform context root/servlet/AJAXProxy`

Configuring settings for IBM Distributed Marketing

If you plan to schedule IBM Distributed Marketing, you must configure an Access Control List (ACL) policy in Tivoli as follows.

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 the following as un-protected objects. Use the complete path in Tivoli, starting from WebSEAL. The Distributed Marketing context root is typically `collaborate`.
 - `WebSEAL junction/Distributed Marketing context root/affiniumcollaborate.jsp`
 - `WebSEAL junction/Distributed Marketing context root/services/CollaborateIntegrationServices1.0`
 - `WebSEAL junction/Distributed Marketing context root/flowchartRunNotifyServlet`
 - `WebSEAL junction/Distributed Marketing context root/js/js_messages.jsp`

- *WebSEAL junction/Distributed Marketing context root/js/format_symbols.jsp*
- *WebSEAL junction/Distributed Marketing context root/alertsService*

Configuring settings for IBM Interact

If you plan to us Interact, you must configure an Access Control List (ACL) policy in Tivoli as follows.

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 the following as un-protected objects. Use the complete path in Tivoli, starting from WebSEAL. The Interact context root is typically Campaign/interact and the Campaign context root is typically Campaign.
 - *WebSEAL junction/Interact context root/flowchartEventPatterns.udo*
 - *WebSEAL junction/Interact context root/saveFlowchartAction.udo*
 - *WebSEAL junction/Interact context root/testRunFlowchart.udo*
 - *WebSEAL junction/Interact context root/getProfileDataAction.udo*
 - *WebSEAL junction/Interact context root/manageIPB.udo*
 - *WebSEAL junction/Campaign context root/initOfferListResolution.udo*
 - *WebSEAL junction/Campaign context root/getOfferListResolutionStatus.udo*

Configuring settings for IBM SPSS Modeler Advantage Marketing Edition

If you plan to us IBM SPSS Modeler Advantage Marketing Edition, you must configure an Access Control List (ACL) policy in Tivoli as follows.

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 the following as an un-protected object. Use the complete path in Tivoli, starting from WebSEAL. The Marketing Platform context root is typically `unica`.
WebSEAL junction/Marketing Platform context root/rest/spssUser

How to integrate IBM EMM with a web access control platform

Topics in this section describe how to integrate IBM EMM with a web access control platform.

Configuration process checklist (Web access control integration)

Integrating IBM EMM with a web access control system is a multi-step process. The following procedure provides an overview of the process, which is described in detail elsewhere in this guide.

About this task

1. "Perform LDAP integration"
Follow instructions for LDAP integration, stopping at the "Test synchronization" step.
2. "Configure web access control integration in IBM EMM"
Set web access control integration properties on the Configuration page.
3. "Restarting the web application server" on page 88
This step is required to ensure that all of your changes are applied.
4. "Test web access control synchronization and IBM EMM login" on page 111
Verify that users and groups synchronize correctly in your web access control system and that you can log in to IBM EMM.

Perform LDAP integration

Perform all of the steps required for LDAP integration as described elsewhere in this guide.

Configure web access control integration in IBM EMM

On the Configuration page, set values of the properties as described in the following table. For complete details on these properties, see the online help on the Configuration page.

About this task

Property	Value
IBM EMM Platform Security Login method details	Select Web access control.
IBM EMM 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 Tivoli Access Manager is \w*

Property	Value
IBM EMM 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 <code>sm_user</code> , and Tivoli Access Manager uses <code>iv-user</code> . For Tivoli Access Manager, set this value to the user name component of the IBM Raw string, not the IBM HTTP string.
IBM EMM General Navigation IBM Marketing Platform URL	<p>Set to <code>http://sm_host:sm_port/sm_realm/unica</code> where</p> <ul style="list-style-type: none"> • <code>sm_host</code> is the name of the machine on which SiteMinder is installed • <code>sm_port</code> is the SiteMinder port number • <code>sm_realm</code> is the SiteMinder realm

Restarting the web application server

Restart the web application server to ensure that all of your configuration changes are applied.

Test web access control synchronization and IBM EMM login

Follow this procedure to test your integration.

Procedure

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 EMM 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 EMM without being presented with the IBM EMM login screen.
4. Use the following guidelines to resolve problems when your web access control software is Netegrity SiteMinder.
 - If you see an IBM EMM 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 EMM, 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 EMM URLs in SiteMinder.
 - If you can access IBM EMM, 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.

Chapter 11. Managing alerts and notifications

IBM Marketing Platform provides support for system alerts and user notifications sent by IBM EMM 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.

See the following topics for more information.

- “Alert and notification subscriptions”
- “Setting system alert and notification subscriptions”
- “Configuring email subscriptions” on page 114

Within Marketing Platform, the IBM EMM Scheduler uses the notification feature, as described in “Schedule notifications” on page 67

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 EMM. Users cannot control these by changing their subscriptions.

Setting system alert and notification subscriptions

Non-administrative users can set their own subscriptions for system alerts and notifications by following this procedure

Procedure

1. Log in to IBM EMM and select **Settings > Users**.
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 subscriptions

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.

About this task

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.

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.
See “Adding a user data source” on page 8 for instructions.
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 EMM as a user with administrative privileges in the 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.
For instructions on how to set the properties, you can use the online help, or see the following topics.

 - “General | Communication | Email” on page 189
 - “Platform | Notifications” on page 212

Chapter 12. SSL implementation

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

URLs that connect using SSL start with HTTPS instead of HTTP.

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 EMM 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.

This section describes one-way SSL in IBM EMM.

About 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 EMM application such as the Campaign listener, needs to serve up a certificate.

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 see 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 EMM

IBM EMM application components can act as either the client or the server in a communication, depending on the situation.

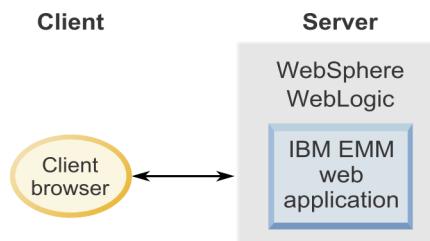
Most IBM EMM applications consist of two parts.

- 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.

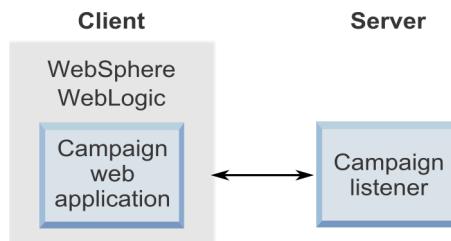
Example 1 - Communication between a browser and an IBM EMM web application

When users communicate with IBM EMM web applications through a browser, the browser is the client and the IBM EMM web application is the server.



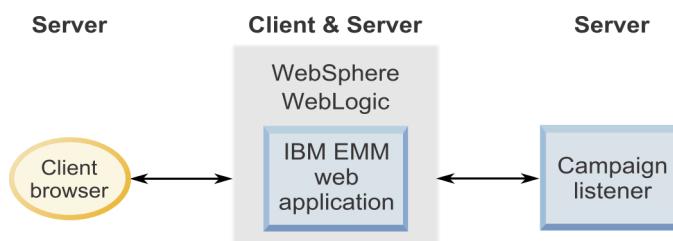
Example 2 - Communication between components of one IBM EMM application

The two components of a single IBM EMM 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 EMM components playing both roles

An IBM EMM 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 EMM

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.

Remember, 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.

Application servers provide default certificates, which require no additional configuration. The application server default certificate is used when you simply enable an SSL port in the application server and do not perform any additional configuration in the application server.

If you use a certificate other than the default certificate supplied by the application server, additional configuration is required. This configuration is described in "Configure your web application servers for SSL" on page 121

C++ component acting as a server

The Campaign listener, Contact Optimization server component, and Attribution Modeler listener are written in C++, and require a certificate stored in PEM format.

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 OpenSSL implementation, no truststore is needed. The Campaign listener, Contact Optimization server component, Attribution Modeler listener, 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 it is the correct format (that is JKS for Java components and PEM for C++ components). 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.

Examples in this chapter show you how to create self-signed certificate files for use with Java and C++ IBM components.

How to implement SSL in IBM EMM

Topics in this section describe how to implement SSL in IBM EMM.

Configuration process checklist (SSL)

Configuring SSL in IBM EMM is a multi-step process. The following procedure provides an overview of the process, which is described in detail elsewhere in this chapter.

About this task

1. “Obtain or create certificates”

Obtain or create certificates if you prefer not to use the default certificates provided by IBM and your application server.

2. “Configure your web application servers for SSL” on page 121

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.

3. “Configure IBM EMM for SSL” on page 122

Set configuration properties in IBM EMM.

4. “Verifying your SSL configuration” on page 128

Log in to each of your IBM EMM applications.

Obtain or create certificates

You can obtain or create certificates in several ways.

About this task

- You can use the default certificates provided by your application server.
- You can create self-signed certificates as described in this section.
- You can obtain certificates from a certificate authority (CA) as described in this section.

How to create self-signed certificates

Use the procedures in this section to create self-signed certificate files for use with IBM EMM.

About this task

- “Creating a certificate for a C++ applicationIBM EMM components”
- “Creating a certificate for Java IBM EMM components” on page 120

Creating a certificate for a C++ applicationIBM EMM components

Follow this procedure to create a certificate for a C++ application.

About this task

The Campaign listener implements SSL using the OpenSSL library. The OpenSSL distribution includes a command-line program called `openssl` that can create a certificate file. For complete details on using this program, consult the OpenSSL documentation or access the help by entering `-help` when you run the program.

Use the following procedure to create a self-signed certificate that you can use when configuring a C++ IBM EMM component for SSL.

Procedure

1. Run `openssl` at the command line.

This program and its associated configuration file, `openssl.cnf`, are included in the `bin` directory of the Campaign installation. It is also available with the OpenSSL distribution.

2. Generate a key. The following example command creates a key named `key.pem`.
`genrsa -out key.pem 1024`
3. Generate a request.

The following example command creates a request named `request.pem`.

```
req -new -key key.pem -out request.pem
```

The tool asks you a series of questions. If you enter a period (.) the field is left blank. For a self-signed certificate, you must at least enter the Common Name.

If you are using the `openssl` tool from the Campaign/bin directory, add the `-config` parameter with a value that points to the `openssl.cnf` file in the same directory. For example:

```
req -config openssl.cnf -new -key key.pem -out request.pem
```

4. Generate a certificate.

The following example command creates a certificate named `certificate.pem` with an expiration of 10,000 days from the day it was created, using the `request.pem` and `key.pem` files.

```
req -x509 -key key.pem -in request.pem -days 10000 -out certificate.pem
```

If you are using the `openssl` tool from the Campaign/bin directory, add the `-config` parameter with a value that points to the `openssl.cnf` file in the same directory. For example:

- ```
req -config openssl.cnf -x509 -key key.pem -in request.pem -days 10000
-out certificate.pem
```
5. Using a text editor, copy the contents of your key and certificate into a new file with a .pem extension.

## Creating a certificate for Java IBM EMM components

Follow this procedure to create a certificate for Java IBM EMM components.

### About this task

IBM EMM web application components written in Java use the JSSE library. The Sun JDK includes a program called keytool that can create a certificate file. Consult the Java documentation for complete details on using this program, or access the help by entering `-help` when you run the program.

Use the following procedure to create a self-signed certificate that you can use when configuring a Java IBM EMM component for SSL.

### Procedure

1. Run keytool at the command line.  
This program is included in the bin directory of the Sun Java JDK.
2. Generate an identity keystore.  
The following example command creates a keystore named UnicaClientIdentity.jks.

```
keytool -genkey -alias UnicaClientIdentity -keyalg RSA -keystore UnicaClientIdentity.jks -keypass clientPwd -validity 1000 -dname "CN=hostName, O=myCompany" -storepass clientPwd
```

Note the following.
  - Make a note of the `-storepass` value (clientPwd in the example) as you need it when you configure the application server.
  - Make a note of the `-alias` value (UnicaClientIdentity in the example) as you need it for the rest of this procedure.
  - The common name (CN) in the distinguished name should be the same as the host name used to access IBM EMM. For example, if the URL for IBM EMM is `https://hostName.companyDomain.com:7002/unica/jsp`, then the CN should be `hostName.companyDomain.com`. The CN portion of the distinguished name is the only required portion; Organization (O) and Organizational Unit (OU) are not required.
  - For WebSphere 6.0, the keystore password and key password must be the same.
3. Generate a certificate based on the identity keystore you created.  
The following example command creates a certificate named UnicaCertificate.cer.

```
keytool -export -keystore UnicaClientIdentity.jks -storepass clientPwd -alias UnicaClientIdentity -file UnicaCertificate.cer
```

The value of `-alias` is the alias you set for the identity keystore (UnicaClientIdentity in the example).
4. Generate a trusted keystore based on the certificate you created.  
The following example command creates a trusted keystore named UnicaTrust.jks.

```
keytool -import -alias UnicaClientIdentity -file UnicaCertificate.cer -keystore UnicaTrust.jks -storepass trustPwd
```

Note the following.

- Type `Y` when prompted to trust the certificate.
- The value of `-alias` is the alias you set for the identity keystore (`UnicaClientIdentity` in the example).
- Make a note of the `-storepass` value (`trustPwd` in the example) as you need it when you configure the application server.

## How to obtain signed certificates

You can use the OpenSSL and keytool programs 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.

### About this task

Note the following.

- For IBM EMM applications written in C++, obtain a certificate in PEM format.
- For all other IBM EMM applications, obtain a certificate in JKS format.

Consult your certificate authority documentation for instructions on how to obtain a signed certificate.

## Configure your web application servers for SSL

On every application server on which an IBM EMM application is deployed, configure the web application server to use the certificates you have decided to employ.

### About this task

See your web application server documentation for details on performing these procedures.

## 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 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.

### Setting the flags in WebSphere

You set the secure and `httponly` flags in the WebSphere administrative console, as follows.

**Tip:** See the WebSphere documentation for complete details.

At the application level for Marketing Platform, do the following.

1. 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.

## Configure IBM EMM for SSL

To configure IBM EMM applications to use SSL, you must set some configuration properties. Use the procedures in this section that are appropriate for your installation of IBM EMM products and the communications that you want to secure using SSL.

### About this task

When you access your IBM EMM 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”
- “Configuring SSL in Marketing Platform with LDAP integration” on page 123
- “Configuring SSL in Marketing Platform with data filters” on page 123
- “Configuring SSL in Interaction History” on page 124
- “Configuring SSL in Attribution Modeler” on page 124
- “Configuring SSL in Marketing Operations” on page 125
- “Configuring SSL in Campaign” on page 125>
- “Configuring SSL in Contact Optimization” on page 126
- “Configuring SSL in Interact” on page 127
- “Configuring SSL in Distributed Marketing” on page 127
- “Configuring SSL in Reports” on page 127
- “Configuring SSL in Digital Analytics for On Premises” on page 127

### Configuring SSL in Marketing Platform

Follow this procedure to configure SSL in Marketing Platform.

#### Procedure

1. Log in to IBM EMM 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 EMM 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 EMM 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.”

## Configuring SSL in Marketing Platform with LDAP integration

Follow this procedure to configure SSL in Marketing Platform.

### Procedure

1. Perform the procedure described in “Configuring SSL in Marketing Platform” on page 122 if you have not done so already.
  2. Log in to IBM EMM and click **Settings > Configuration**.  
The Configuration page appears.
  3. Navigate to the IBM EMM | 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 EMM | 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.

## 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.

## **Procedure**

1. Perform the procedure described in “Configuring SSL in Marketing Platform” on page 122 if you have not done so already.
2. Obtain the following.
  - A copy of the certificate file you created in “Obtain or create certificates” on page 118
  - 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.

## **Configuring SSL in Interaction History**

Follow this procedure to configure SSL in Interaction History.

## **Procedure**

1. Log in to IBM EMM and click **Settings > Configuration**.
2. Set the value of the **Interaction History | navigation | HTTPS Port** property to the https port in the application server on which Interaction History is deployed.
3. Set the value of the **Interaction History | navigation | Server URL** property to the Interaction History URL.

For example: `https://host.domain:SSL_port/unica`  
where:

  - *host* is the name or IP address of the machine on which Interaction History is installed
  - *domain* is your company domain in which your IBM EMM products are installed
  - *SSL\_Port* is the SSL port in the application server on which Interaction History is deployed

Note the https in the URL.

## **Configuring SSL in Attribution Modeler**

Follow this procedure to configure SSL in Attribution Modeler

## **Procedure**

1. Log in to IBM EMM and click **Settings > Configuration**.

The Configuration page appears.
2. Set the value of the **Attribution Modeler | navigation | httpsPort** property to the https port in the application server on which Attribution Modeler is deployed.
3. Check the **Attribution Modeler | navigation | serverURL** property to ensure that it uses https in the URL.

4. Ensure that the value of the Attribution Modeler | AMListener | serverPort property is an appropriate port for SSL.
5. Set the value of the Attribution Modeler | AMListener | useSSL property to TRUE.

## Configuring SSL in Marketing Operations

Follow this procedure to configure SSL in Marketing Operations.

### Procedure

1. Log in to IBM EMM 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.

### Procedure

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 value of `unicaServerSSLFile` to the full path of the PEM file you are using. (The file that IBM provides, `unicaclient.pem`, is located in the `security` directory.) For example:  
`unicaServerSSLFile=C:/Unica/security/certificateFile.pem`
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 Contact Optimization

Follow this procedure to configure SSL in Contact Optimization.

### Procedure

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 PEM file you are using. (The file that IBM provides, `unicaclient.pem`, is located in the `security` directory of your Contact Optimization installation.)
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 `ACOptAdmin`, perform the following steps.
  - a. Obtain the following.
    - A copy of the certificate file you created in “Obtain or create certificates” on page 118
    - 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 `ACOptAdmin` 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.

### About this task

- 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.  
See “Configuring SSL in Marketing Platform” on page 122 for details.
- 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.

**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.

### Procedure

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 EMM as described in the Cognos documentation.
4. Register the IBM EMM 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.

### About this task

Digital Analytics for On Premises does not communicate with Marketing Platform.

## Verifying your SSL configuration

Follow this procedure to verify your SSL configuration.

### Procedure

1. Start each of your IBM EMM applications.
2. Log in to IBM EMM and access each of your installed IBM EMM 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 EMM 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 EMM.

- OpenSSL documentation - <http://www.openssl.org/docs/>
- keytool documentation - <http://download.oracle.com/javase/1.4.2/docs/tooldocs/windows/keytool.html>
- List of certificate authorities - [http://www.dmoz.org/Computers/Security/Public\\_Key\\_Infrastructure/PKIX/Tools\\_and\\_Services/Third\\_Party\\_Certificate\\_Authorities/](http://www.dmoz.org/Computers/Security/Public_Key_Infrastructure/PKIX/Tools_and_Services/Third_Party_Certificate_Authorities/)

---

## Chapter 13. Data filter configuration

The various IBM EMM 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.

In general, when an IBM EMM application uses data filtering, IBM EMM administrators can specify data access restrictions in IBM EMM products based on configurable data filters. Data filters make it possible to restrict the customer data that an IBM EMM user can view and work with in IBM EMM 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.

---

### About setting up data filters

The Marketing Platform provides the following functions that IBM EMM 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 EMM 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.

### 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.

## Two ways to create data filters: automatic generation and manual specification

IBM EMM 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).

This method is described in “Configuration process checklists” on page 140.

You might want to use this method if you must create many data filters.

### Manual specification

The `datafilteringScriptTool` utility can create data filters one by one, based on field values that you specify.

This method is described in “Configuration process checklist (manual specification of data filters).”

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.

---

## How to set up data filters using manual specification

Topics in this section describe how to set up data filters using manual specification.

### Configuration process checklist (manual specification of data filters)

Configuring data filters using the manual specification method is a multi-step process. The following procedure provides an overview of the process, which is described in detail elsewhere in this guide.

#### About this task

1. “Plan your data filter criteria (manual generation)” on page 131  
Decide what customer data you want to secure.
2. “Obtain required information (manual specification)” on page 131  
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.
3. “Create the XML to specify data filters (automatic generation)” on page 142  
Create the XML file that specified the customer data used as criteria in each data filter.

4. “Populate the data filter system tables” on page 132

Run the `datafilteringScriptTool` utility, which uses your XML to populate the Marketing Platform system tables that are used for data filters.

5. “Assign users and groups to data filters” on page 132

Use the IBM EMM data filter user interface to perform searches for users, groups, and data filters and then select items from the search results and assign them.

## Install Marketing Platform

Install Marketing Platform. Perform all of the required steps described in the installation guide.

### About this task

## Plan 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.

### About this task

For example, you might want to restrict access to customer data based on the geographical sales territory to which the IBM EMM 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 EMM 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).

## Obtain required information (manual specification)

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.

### About this task

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

## Create the XML to specify data filters (manual specification)

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.

## Populate the data filter system tables

Run the `datafilteringScriptTool` utility, which uses your XML to populate the data filter system tables.

### About this task

See “`datafilteringScriptTool`” on page 172 for details on using the utility.

**Note:** If you need to delete data filters, run the `ManagerSchema_PurgeDataFiltering.sql` script as described in “`ManagerSchema_PurgeDataFiltering.sql`” on page 181.

## Assign users and groups to data filters

Use the IBM EMM data filter user interface to perform searches for users, groups, and data filters and then select items from the search results and assign them. You can also perform searches to view data filters that have already been assigned to users and groups.

## Data filter XML reference (manual specification)

This section describes the XML elements for which you must provide values when you use the `datafilteringScriptTool` to generate data filters by specifying them manually.

### 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.

| 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<br>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.

| 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<br>Field: logical_name |
| type    | Data type of this field in the customer table. Allowed values are: <ul style="list-style-type: none"><li>• java.lang.String</li><li>• java.lang.Long</li><li>• java.lang.Double</li><li>• java.lang.Boolean</li><li>• java.lang.Date (The date format is month/day/year, where the month, day, and year are all expressed as numbers.)</li></ul> | Table: df_logical_field<br>Field: type         |

### AddDataTable | dataTable

This group of elements is used to assign IDs to customer tables.

| 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<br>Field: table_name |

## 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.

| 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<br>Field: physical_name |
| logicalFieldId | ID of the logical field in the AddLogicalFields   logicalFields   LogicalField category.                                   | N/A                                           |

## AddDataFilters | dataFilters | DataFilter

This group of elements is used to create a data filter.

| Element  | Description                                                                                                                 | System table |
|----------|-----------------------------------------------------------------------------------------------------------------------------|--------------|
| configId | ID of the data configuration in the AddDataConfiguration   dataConfiguration category 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.

| Element        | Description                                                                                                                                                                                      | System table                                    |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------|
| logicalFieldId | ID of the logical field in the AddLogicalFields   logicalFields   LogicalField category.                                                                                                         | 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_field_constraint<br>Field: expression |

## 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.

| 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<br>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.

| Element        | Description                                                                                                                                                                                  | System table |
|----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|
| logicalFieldId | ID of the logical field in the AddLogicalFields   logicalFields   LogicalField category. 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.

| Element    | Description                                                                                                                                                                                                                                                                     | System table |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|
| audienceId | ID of the audience to be used in this association. Must be an ID value in an AddAudience   audience category.                                                                                                                                                                   | N/A          |
| tableId    | ID of the table to be used in this association. Must be an ID value in an AddDataTable   dataTable category. 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 category.                                                                                                                                       | N/A          |

## 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 9. Territory table fields*

| Fields<br>(physical name) | Fields<br>(name in Campaign) | Data                                                                                                                                                           | Data type        |
|---------------------------|------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|
| cust_region               | CustomerRegion               | <ul style="list-style-type: none"> <li>• Africa</li> <li>• Africa</li> <li>• Asia</li> <li>• Europe</li> <li>• Middle East</li> <li>• North America</li> </ul> | 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**.

```
<?xml version="1.0" encoding="UTF-8"?>
<ExecuteBatch>
<name>SeedData</name>
<operations>
<!-- Create the data configuration that groups related Data Filters -->
<ExecuteBatch>
<name>DataFilters</name>
<operations>
<AddDataConfiguration>
<dataConfiguration>
<id>1</id>
<name>Territory</name>
</dataConfiguration>
</AddDataConfiguration>
</operations>
</ExecuteBatch>
```

```

<!-- Add logical fields used to define data filters -->
<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>

<!-- Add the 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>

<!--Add Audiences-->
<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 EMM with an account that has Admin access in Marketing Platform.

He knows that groups have already been set up in IBM EMM 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 EMM 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.

---

## **How to set up data filters using automatic specification**

Topics in this section describe how to set up data filters using automatic specification.

## Configuration process checklists

Configuring data filters using the automatic generation method is a multi-step process. The following procedure provides an overview of the process, which is described in detail elsewhere in this guide.

### About this task

1. "Plan your data filter criteria (automatic generation)"  
Decide what customer data you want to secure.
2. "Obtain the JDBC driver for your database" on page 141  
Obtain the Type 4 JDBC driver that provides connectivity to the database containing the table on which you want to base your data filters.
3. "Obtain required information (automatic generation)" on page 141  
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.
4. "Create the XML to specify data filters (automatic generation)" on page 142  
Create the XML file that specified the customer data used as criteria in each data filter.
5. "Populate the data filter system tables" on page 132  
Run the `datafilteringScriptTool` utility, which uses your XML to populate the Marketing Platform system tables that are used for data filters.
6. "Assign users and groups to data filters" on page 132  
Use the IBM EMM data filter user interface to perform searches for users, groups, and data filters and then select items from the search results and assign them.

## Install Marketing Platform

Install Marketing Platform. Perform all of the required steps described in the installation guide.

### About this task

## Plan 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.

### About this task

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 EMM 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 EMM 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, 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 EMM 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.

## Obtain the JDBC driver for your database

A JDBC driver is required by the data filter generation utility (datafilteringScriptTool) when you use it to generate data filters automatically.

### About this task

#### Procedure

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.

## Obtain required information (automatic generation)

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.

### About this task

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.

## Create the XML to specify data filters (automatic generation)

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.

## Populate the data filter system tables

Run the `datafilteringScriptTool` utility, which uses your XML to populate the data filter system tables.

### About this task

See “`datafilteringScriptTool`” on page 172 for details on using the utility.

**Note:** If you need to delete data filters, run the `ManagerSchema_PurgeDataFiltering.sql` script as described in “`ManagerSchema_PurgeDataFiltering.sql`” on page 181.

## Assign users and groups to data filters

Use the IBM EMM data filter user interface to perform searches for users, groups, and data filters and then select items from the search results and assign them. You can also perform searches to view data filters that have already been assigned to users and groups.

## Data filter XML reference (automatic generation)

This section describes the XML elements for which you must provide values when you use the `datafilteringScriptTool` to generate data filters automatically.

### 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.

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.

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: <ul style="list-style-type: none"><li>• java.lang.String</li><li>• java.lang.Long</li><li>• java.lang.Double</li><li>• java.lang.Boolean</li><li>• java.lang.Date (The date format is month/day/year, where the month, day, and year are all expressed as numbers.)</li></ul>	Table: df_logical_field Field: type

### GenerateDataFilters

This group of elements is used to generate data filters.

Element	Description	System table
tableName	Physical name of the table from which you want to generate data filters. If the database is case-sensitive, must match case used in the database	Table: df_table Field: table_name

Element	Description	System table
configurationName	Name of the data configuration in the AddDataConfiguration   dataConfiguration category 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.

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 category. 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
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.

Element	Description	System table
logicalFieldName	Name of the logical field in the AddLogicalFields   logicalFields   LogicalField category.	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.

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

## 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.

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 category.	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.

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.

Element	Description	System table
logicalFieldId	ID of the logical field in the AddLogicalFields   logicalFields   LogicalField category. 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.

Element	Description	System table
audienceId	ID of the audience to be used in this association. Must be an ID value in an AddAudience   audience category.	N/A
tableId	ID of the table to be used in this association. Must be an ID value in an AddDataTable   dataTable category. 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 category.	N/A

### **AddDataFilters | dataFilters | DataFilter**

This group of elements is used to create a data filter.

Element	Description	System table
configId	ID of the data configuration in the AddDataConfiguration   dataConfiguration category 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.

Element	Description	System table
logicalFieldId	ID of the logical field in the AddLogicalFields   logicalFields   LogicalField category.	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_field_constraint Field: expression

## 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 10. Geographic table fields*

Fields (Physical name)	Fields (Name in Campaign)	Data	Data type
country	Country	<ul style="list-style-type: none"> <li>• USA</li> <li>• France</li> <li>• Britain</li> </ul>	java.lang.String

*Table 10. Geographic table fields (continued)*

Fields (Physical name)	Fields (Name in Campaign)	Data	Data type
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**.

```
<?xml version="1.0" encoding="UTF-8"?>
<ExecuteBatch>
<name>SeedData</name>
<operations>
<!-- Create the data configuration that groups related Data Filters -->
<ExecuteBatch>
<name>DataFilters</name>
<operations>
<AddDataConfiguration>
<dataConfiguration>
<id>1</id>
<name>Geographic</name>
</dataConfiguration>
</AddDataConfiguration>
</operations>
</ExecuteBatch>
<!-- Add logical fields used to define data filters -->
<AddLogicalFields>
<logicalFields>
<LogicalField>
<id>1</id></pre>

```

```

<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>

<!-- Provide information needed to 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>
</fixedFields>
<fixedFields>
<FixedField>
<expression>France</expression>
<logicalFieldName>Country</logicalFieldName>
<physicalFieldName>country</physicalFieldName>
</FixedField>
</fixedFields>
<fixedFields>
<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>

<!--Add Audiences-->
<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>
 </audience>
 </AddAudience>
 </operations>
</ExecuteBatch>

```

```

<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>
</pre>

```

## 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.

---

## **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.

### **About this task**

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.

Once you have created the XML, populate the data filter tables and assign users and groups as described in this guide.

---

## Chapter 14. Data filter management

IBM EMM administrators can specify data access restrictions in IBM EMM products based on configurable data filters. Data filters make it possible to restrict the customer data that an IBM EMM user can view and work with in IBM EMM applications.

To work with data filters in the **Settings > Data Filters** pages, the following must be true.

- The data filters must be set up in Marketing Platform system tables, as described in Chapter 13, “Data filter configuration,” on page 129.
- You must log in as a user with the **Administer Data Filters** page permission. By default, the **AdminRole** role has this permission.

---

### Restricting data access through user and group assignments

To restrict data access for individual users or groups of users, you assign them to data filters.

All users and groups that exist in IBM EMM 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 to multiple data filters.

**Note:** Groups do not acquire the data filter assignments of their parent groups.

---

### About advanced search

IBM EMM 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."

---

## 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 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.

---

## Data filter assignments

This section describes how configure data filters and manage data filter assignments.

### Viewing assigned data filters

Use this procedure to view assigned data filters

#### Procedure

1. Log in to IBM EMM 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.

#### 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.

## **Procedure**

1. Log in to IBM EMM 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**.

## **Results**

The selected users and groups are assigned to the selected data filters.

## **Removing data filter assignments**

Use this procedure to remove data filter assignments.

## **Procedure**

1. Log in to IBM EMM 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**.

## **Results**

The selected assignments are deleted. The data filters themselves are not deleted.



---

## Chapter 15. IBM Marketing Platform logs

You can use the system log to track usage and detect potential security problems. The system log can help you detect erroneous or malicious behavior as it occurs.

---

### About the system log

You should check the system log first if the Marketing Platform application malfunctions or if you think a break-in might have occurred or been attempted.

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

The first part of the system log shows the configuration settings that are read into the system from the `uasm.conf` configuration file on startup. Viewing the configuration settings in the log file is an easy way to check settings that control properties for IBM EMM passwords, the Marketing Platform authentication data store, the Marketing Platform web server root, and the system log and system audit trail.

**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.

### 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.
- 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>.



---

## Chapter 16. Configuration process checklists

Configuring data filters using the automatic generation method is a multi-step process. The following procedure provides an overview of the process, which is described in detail elsewhere in this guide.

### About this task

1. “Plan your data filter criteria (automatic generation)” on page 140  
Decide what customer data you want to secure.
2. “Obtain the JDBC driver for your database” on page 141  
Obtain the Type 4 JDBC driver that provides connectivity to the database containing the table on which you want to base your data filters.
3. “Obtain required information (automatic generation)” on page 141  
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.
4. “Create the XML to specify data filters (automatic generation)” on page 142  
Create the XML file that specified the customer data used as criteria in each data filter.
5. “Populate the data filter system tables” on page 132  
Run the `datafilteringScriptTool` utility, which uses your XML to populate the Marketing Platform system tables that are used for data filters.
6. “Assign users and groups to data filters” on page 132  
Use the IBM EMM data filter user interface to perform searches for users, groups, and data filters and then select items from the search results and assign them.

---

## Configuration process checklist (manual specification of data filters)

Configuring data filters using the manual specification method is a multi-step process. The following procedure provides an overview of the process, which is described in detail elsewhere in this guide.

### About this task

1. “Plan your data filter criteria (manual generation)” on page 131  
Decide what customer data you want to secure.
2. “Obtain required information (manual specification)” on page 131  
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.
3. “Create the XML to specify data filters (automatic generation)” on page 142  
Create the XML file that specified the customer data used as criteria in each data filter.
4. “Populate the data filter system tables” on page 132  
Run the `datafilteringScriptTool` utility, which uses your XML to populate the Marketing Platform system tables that are used for data filters.
5. “Assign users and groups to data filters” on page 132

Use the IBM EMM data filter user interface to perform searches for users, groups, and data filters and then select items from the search results and assign them.

## Configuration process roadmap (Active Directory integration)

Use this configuration process roadmap to scan the tasks required to integrate IBM EMM with Windows Active Directory. The Topic column provides links to the topics that describe the tasks in detail.

Topic	Information
"Required information" on page 82	Obtain information about your Windows Active Directory server, which is required for integration with IBM EMM.
"Group membership, mapping, and application access" on page 84	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 84	If your directory server does not allow anonymous access (the most common configuration), configure an IBM EMM user account to hold a directory server administrator user name and password.
<ul style="list-style-type: none"> <li>“Setting connection properties” on page 85</li> <li>“Setting LDAP synchronization properties” on page 85</li> <li>“Setting user attributes map properties” on page 86</li> <li>“Mapping LDAP groups to IBM EMM groups” on page 86</li> </ul>	Configure the Marketing Platform for integration by setting values on the Configuration page.
"Testing synchronization" on page 87	Verify that users are imported as expected, and if you are using group based synchronization, verify that users and groups are synchronizing properly.
"Setting up an Active Directory user with PlatformAdminRole permissions" on page 87	Set up administrator access to Marketing Platform, required when Windows integrated login is enabled.
"Setting the security mode to Windows Integrated Login" on page 88	Set the security mode values on the Configuration page.
"Configuring Internet Explorer" on page 88	Set a custom security level in every instance of Internet Explorer that is used to access IBM EMM. This is required with Windows integrated login, to prevent users from being presented with the IBM EMM login screen.
"Restarting the web application server" on page 88	This step is required to ensure that all of your changes are applied.
"Testing login as an Active Directory user" on page 88	Verify that you can log in to IBM EMM as an Active Directory user.

## Configuration process roadmap (LDAP integration)

Use this configuration process roadmap to scan the tasks required to integrate IBM EMM with LDAP. The Topic column provides links to the topics that describe the tasks in detail.

Topic	Information
"Required information" on page 82	Obtain information about your LDAP server, which is needed for integration with IBM EMM.

Topic	Information
"Group membership, mapping, and application access" on page 84	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 84	If your directory server does not allow anonymous access (the most common configuration), configure an IBM EMM user account to hold a directory server administrator user name and password.
<ul style="list-style-type: none"> <li>• "Setting connection properties" on page 85</li> <li>• "Setting LDAP synchronization properties" on page 85</li> <li>• "Setting user attributes map properties" on page 86</li> <li>• "Mapping LDAP groups to IBM EMM groups" on page 86</li> </ul>	Configure the Marketing Platform for integration by setting values on the Configuration page.
"Testing synchronization" on page 87	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 99	Set the security mode values on the Configuration page.
"Restarting the web application server" on page 88	This step is required to ensure that all of your changes are applied.
"Testing login as an LDAP user" on page 99	Verify that you can log in to IBM EMM as an LDAP user.

## Configuration process checklist (Web access control integration)

Integrating IBM EMM with a web access control system is a multi-step process. The following procedure provides an overview of the process, which is described in detail elsewhere in this guide.

### About this task

1. "Perform LDAP integration" on page 110  
Follow instructions for LDAP integration, stopping at the "Test synchronization" step.
2. "Configure web access control integration in IBM EMM" on page 110  
Set web access control integration properties on the Configuration page.
3. "Restarting the web application server" on page 88  
This step is required to ensure that all of your changes are applied.
4. "Test web access control synchronization and IBM EMM login" on page 111  
Verify that users and groups synchronize correctly in your web access control system and that you can log in to IBM EMM.

## Configuration process checklist (SSL)

Configuring SSL in IBM EMM is a multi-step process. The following procedure provides an overview of the process, which is described in detail elsewhere in this chapter.

### About this task

1. "Obtain or create certificates" on page 118  
Obtain or create certificates if you prefer not to use the default certificates provided by IBM and your application server.
2. "Configure your web application servers for SSL" on page 121

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.

3. “Configure IBM EMM for SSL” on page 122  
Set configuration properties in IBM EMM.
4. “Verifying your SSL configuration” on page 128  
Log in to each of your IBM EMM applications.

---

# Chapter 17. 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 167 - registers alerts and configurations for IBM EMM products
- “`configTool`” on page 168 - imports, exports, and deletes configuration settings, including product registrations
- “`datafilteringScriptTool`” on page 172 - creates data filters
- “`encryptPasswords`” on page 173 - encrypts and stores passwords
- “`partitionTool`” on page 174 - creates database entries for partitions
- “`populateDb`” on page 176 - populates the Marketing Platform database
- “`restoreAccess`” on page 177 - restores a user with the `platformAdminRole` role
- “`scheduler_console_client`” on page 179 - lists or starts IBM EMM 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.

## 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.

### **Before you begin**

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.

## Procedure

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 EMM 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 EMM 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 EMM 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 EMM 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 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 EMM 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 EMM.
- 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 EMM, 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 11. 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

Table 11. Product names for configTool registration and unregistration (continued)

Product name	Name used in configTool
CustomerInsight	Insight
Digital Analytics for On Premises	NetInsight
Opportunity Detection	Detect
Leads	Leads
Interaction History	InteractionHistory
Attribution Modeler	AttributionModeler
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 productName -f app_config.xml -o
• Unregister an application named productName.
configTool -u productName
```

## **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).

```
:call exec

"%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`.

```
:call exec

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"
```

## 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.

See Chapter 17, “IBM Marketing Platform utilities and SQL scripts,” on page 165 for additional prerequisites.

## 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 Platform was 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 EMM 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
```

## 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.

See Chapter 17, “IBM Marketing Platform utilities and SQL scripts,” on page 165 for additional restrictions.

## 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 EMM 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 *partitionNameAdminUser*.

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"
```

## populateDb

The populateDb utility inserts default (seed) data in the Marketing Platform system tables.

The IBM EMM 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
```

## 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 Active Directory integration

If you implement Windows Active Directory integration 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 from Windows integrated login 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 *loginName***

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
```

## scheduler\_console\_client

Jobs configured in the IBM EMM 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
```

- Execute jobs configured to listen for a trigger named trigger1.

```
scheduler_console_client -s -t trigger1
```

---

## 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 as described in “populateDb” on page 176. 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, as described in “configTool” on page 168.
- 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.

---

## **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.

---

## **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, as described in “SQL scripts for creating system tables.”
- Run the populateDB utility as described in “populateDb” on page 176. 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, as described in “configTool” on page 168.
- 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.

---

## 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.

Datasource Type	Script Names
IBM DB2®	<ul style="list-style-type: none"><li>• ManagerSchema_DB2.sql</li><li>If you plan to support multi-byte characters (for example, Chinese, Japanese, or Korean), use the ManagerSchema_DB2_unicode.sql script.</li><li>• ManagerSchema_DB2_CeateFKConstraints.sql</li><li>• active_portlets.sql</li></ul>
Microsoft SQL Server	<ul style="list-style-type: none"><li>• ManagerSchema_SQLServer.sql</li><li>• ManagerSchema_SQLServer_CeateFKConstraints.sql</li><li>• active_portlets.sql</li></ul>
Oracle	<ul style="list-style-type: none"><li>• ManagerSchema_Oracle.sql</li><li>• ManagerSchema_Oracle_CeateFKConstraints.sql</li><li>• active_portlets.sql</li></ul>

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.

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.



---

## Appendix A. Configuration properties on the Configuration page

This section describes the configuration properties found on the Configuration page.

---

### Marketing Platform configuration properties

This section describes the Marketing Platform configuration properties on the Configuration page.

#### General | Navigation

Properties in this category specify values that are used internally to navigate among IBM EMM 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 EMM 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 EMM 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`.

**Important:** If IBM EMM 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

Not defined

### **Example**

In an environment configured for SSL, the URL might look like this:

`https://machineName.companyDomain.com:8080/unica`

## **General | Data filtering**

Properties in this category specify values used when data filtering is implemented.

### **Default table name**

#### **Description**

In conjunction with Default audience name, determines the set of data filters (that is, the data configuration) from which the data filter user interface in IBM EMM reads filters and assignments.

#### **Default value**

Undefined

#### **Valid Values**

Physical name of the customer table that contains the fields used as data filter criteria. Maximum of 50 characters of type varchar.

### **Default audience name**

#### **Description**

In conjunction with Default table name, determines the set of data filters (that is, the data configuration) from which the data filter user interface in IBM EMM reads filters and assignments.

#### **Default value**

Undefined

#### **Valid Values**

When configuring data filters for Distributed Marketing, the name must be the same as the name given to an audience level in Campaign. Maximum of 50 characters of type varchar.

### **Enable data filter cache**

#### **Description**

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.

#### **Default value**

`False`

## **General | Password settings**

Properties in General|Password category specify the policies that apply to IBM EMM 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 EMM 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 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 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 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 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**

**Valid Values**

Any integer

**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.

**TokenLifetime****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 region****Description**

Specifies the default locale for the Marketing Platform. If you plan to install Campaign, you should set this value to match the locale set for Campaign in Campaign's defaultLocale property.

**Default value**

English

**Valid Values**

Supported locales

**Trusted application enabled****Description**

When this value is set to True, the Marketing Platform must be deployed in an environment that has an SSL port and the IBM Marketing Platform URL property in the General > Navigation category must be set to use https.

**Default value**

False

**Valid Values**

True | False

**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.

### **Default value**

smtp

## **Email Server host**

### **Description**

Specifies the name of the mail server used for sending system alerts and notifications to users.

### **Default value**

localhost

## **Email Server port**

### **Description**

Specifies the port of the mail server used for sending system alerts and notifications to users.

### **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.

### **Default value**

Not defined

## **Authentication required for mail server**

### **Description**

Specifies whether the mail server requires authentication.

### **Default value**

False

## **IBM EMM 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 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 only if your mail server requires authentication.

### **Default value**

emailDS

## **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 setting**

### **Description**

Specifies the locale preference for IBM EMM users. When you set this property on the Configuration page, the setting you apply is the default setting throughout IBM EMM for all users, except those whose locale preference is set individually through the Marketing Platform's 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 EMM applications.

Availability of locales may vary depending on the IBM EMM 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 EMM 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**

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 that 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 EMM. The default is the default dashboard.

#### **Default value**

The default dashboard.

#### **Valid Values**

Any IBM EMM URL except form submissions pages, edit pages, and search result pages.

### **Domain name**

#### **Description**

The name of the domain where IBM EMM is installed. The value is set during installation. You should not change this unless the domain name changes.

#### **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. 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`

## Platform | Scheduler

Properties in this category allow you to enable and tune the performance of the IBM EMM Scheduler.

### Client polling interval

#### Description

Campaign polls the IBM EMM Scheduler for jobs at regular intervals, specified in milliseconds by this value. The default value is 60 seconds. You should not set this property to any value less than 10000 (10 seconds), because this can decrease campaign performance.

#### Default value

`60000`

### Client initialization delay

#### Description

The amount of time, expressed in milliseconds, that the Campaign scheduler thread waits before polling the IBM EMM Scheduler for jobs when Campaign first starts up. Set this value to be at least as long as it takes for 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 > Scheduled Tasks 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.

**Default value**

True

**Valid Values**

True | False

**Platform | Scheduler | Recurrence definitions**

Properties in this category set the recurrence patterns for the IBM EMM 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 \*

## **Platform | Scheduler | Schedule registrations | [Product] | [Object type]**

A different category exists for each of the object types that can be scheduled with the IBM EMM Scheduler. Properties in these categories should not normally be changed.

### **Executor class name**

#### **Description**

The class that the IBM EMM Scheduler uses to trigger a flowchart or mailing run.

#### **Default value**

### **Status polling interval**

#### **Description**

At regular intervals, the IBM EMM Scheduler polls the product for the run status of scheduled objects that have not reported status. The interval is specified here in milliseconds. The default value is 10 minutes. Setting a more frequent polling interval (a smaller value) affects the system performance. Setting a less frequent polling interval (a larger value) reduces the load on the system. For example, you might want to 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

### **Group Name to receive the Job Notifications**

#### **Description**

Notifications for all schedules for each object type can be sent to all members of a group. When you set up these group-based notifications, you enter the names of the group whose members you want to receive the notifications for the object type. For complete information, see “Schedule notifications” on page 67.

## **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 EMM scheduler. 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 configured scheduler groups 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.

## Platform | Security

The property in this category specifies the login mode for IBM EMM products.

### Login method

#### Description

Specifies the authentication mode for all IBM EMM products installed and configured to work together, as follows:

- If you set the value to Windows integrated login, IBM EMM products use Windows Active Directory for authentication.
- If you set the value to IBM Marketing Platform , IBM EMM products use the Marketing Platform for authentication and authorization.
- If you set the value to LDAP, IBM EMM products use an LDAP server for authentication.
- If you set the value to Web access control, IBM EMM products use web access control software for authentication.

#### Default value

IBM Marketing Platform

#### Valid Values

Windows integrated login | IBM Marketing Platform | LDAP | Web access control

## Platform | Security | Login method details | Windows integrated login

Properties in this category are used to configure Windows integrated login.

### Domain

#### Description

Sets the value of JCIFS SMB client library property jcifs.smb.client.Domain. Specifies the domain that is used if no domain is specified in an SMB URL. Set this value to the Windows domain name. For most environments, set either this property or the Domain Controller property.

#### Default value

Undefined.

#### Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory server and Windows integrated login is enabled.

### Client Timeout

#### Description

Sets the value of JCIFS SMB client library property `jcifs.smb.client.soTimeout`. Specifies the amount of time, in milliseconds, before sockets are closed if there is no activity between the client and server. This number should be as small as possible but long enough to allow the protocol handshaking to complete, which depends on network characteristics.

#### Default value

1000

#### Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory server and Windows integrated login is enabled.

### Cache Policy

#### Description

Sets the value of JCIFS SMB client library property `jcifs.netbios.cachePolicy`. Specifies the amount of time, in seconds, that the NetBIOS name is cached to reduce redundant name queries. If the value is set to 0 no caching takes place. If the value is set to -1 the cache is never cleared. This property is used when SMB signing is enabled and required in a Windows 2003 domain.

#### Default value

0

#### Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory server and Windows integrated login is enabled.

### Domain Controller

#### Description

Sets the value of JCIFS SMB client library property `jcifs.http.domainController`. Specifies the IP address of a server that should be used to authenticate HTTP clients (used by NtlmHttpFilter and NetworkExplorer). You may use the IP address of a workstation in the domain specified in the Domain property. For most environments, set either this property or the Domain property.

#### Default value

Undefined.

## **Availability**

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory server and Windows integrated login is enabled.

## **IP of the WINS server**

### **Description**

Sets the value of JCIFS SMB client library property jcifs.netbios.wins. Specifies the IP address of the WINS server. You may enter multiple IP addresses, separated by commas (for example 192.168.100.30, 192.168.100.31). The WINS server is queried to resolve the domain specified in the Domain property to an IP address of a domain controller. This property is required when accessing hosts on different subnet (such as a domain controller by name) and it is highly recommended if a WINS server is available.

### **Default value**

Undefined

### **Availability**

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory server and Windows integrated login is enabled and Windows integrated login is enabled.

## **Strip Domain**

### **Description**

Specifies whether the Marketing Platform removes a domain from users' login names when they access the IBM EMM. If your Windows configuration requires a domain to be included with users' login names when they log in, set this value to False.

### **Default value**

True

### **Valid Values**

True | False

### **Availability**

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory server and Windows integrated login is enabled.

## **Retry on Authentication Failure**

### **Description**

If a user login fails, the system allows another login attempt if this value is set to True. Set to False if you want to disallow more than one login attempt.

### **Default value**

True

### **Valid Values**

True | False

#### Availability

This property is used only when the Marketing Platform is configured to integrate with a Windows Active Directory server and Windows integrated login is enabled.

### 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 EMM user that has been given LDAP administrator login 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 user name you created for the IBM EMM 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 EMM 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

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

## 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 Tivoli Access Manager is \w\*

### **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 Tivoli Access Manager (TAM) uses iv-user. For TAM, 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.

## **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 is 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 EMM 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

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 EMM user that has been given LDAP administrator login credentials.

Set the value of this property to the user name you created for the IBM EMM 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 EMM 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 EMM | 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 12. 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 EMM 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.

## **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.

## **Platform | Notifications**

Properties in this category control system behavior for notifications that IBM EMM 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

---

## IBM Digital Analytics configuration properties

This section describes the IBM Digital Analytics configuration properties on the Configuration page.

These configuration properties are used in configuring single sign-on between IBM Digital Analytics and IBM EMM. See the *IBM Marketing Platform Administrator's Guide* for details about this integration.

### Coremetrics

The property in this category is part of the configuration for enabling single sign-on between IBM Digital Analytics and IBM EMM.

#### Enable Coremetrics Analytics

##### Description

This is part of the configuration for enabling single sign-on between IBM Digital Analytics and IBM EMM.

Set to true as one of the steps for enabling single sign-on.

See the *IBM Marketing Platform Administrator's Guide* for details about this integration.

**Default value**  
false

### Coremetrics | Integration | partitions | partition[n]

Properties in this category are part of the configuration for enabling single sign-on between IBM Digital Analytics and IBM EMM.

#### Platform user for Coremetrics account

##### Description

Specifies the login name of the IBM EMM user account that holds the IBM Digital Analytics shared secret in a data source.

This is part of the configuration for enabling single sign-on between IBM Digital Analytics and IBM EMM. See the *IBM Marketing Platform Administrator's Guide* 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 IBM Digital Analytics shared secret.

This is part of the configuration for enabling single sign-on between IBM Digital Analytics and IBM EMM. See the *IBM Marketing Platform Administrator's Guide* for details about this integration.

**Default value**  
CoremetricsSDS

---

## Interaction History Configuration Properties

Log in to the IBM Enterprise Marketing Management Suite, and then browse to **Settings > Users** to set the Interaction History configuration properties on the Configuration page. You can use all the features of Interaction History after you set the configuration properties.

### Interaction History

The properties that are listed in the **Interaction History** category specify values that are used internally. Normally, these values are set automatically by the installer during installation.

#### ETL Server Name (without domain)

##### Description

The name of the computer on which Interaction History is installed. If for some reason you must set the value manually, use the name of the computer, and not localhost.

**Default value**

localhost

#### Operating System

##### Description

The operating system on which Interaction History is installed.

**Default value**

AIX

##### Valid values

AIX

### Interaction History | navigation

The properties that are listed in the **Interaction History | navigation** category specify values that are used internally to navigate among IBM products.

#### HTTPS Port

##### Description

The **HTTPS Port** specifies the SSL port in the web application server on which Interaction History is deployed. The property is used internally for communication among IBM products, when SSL is enabled.

**Default value**

7001

## HTTP Port

### Description

The **HTTP Port** specifies the HTTP port in the web application server on which Interaction History is deployed. The property is used internally for communication among IBM products.

### Default value

7001

## Server URL

### Description

The **Server URL** specifies the URL that is used for the IBM EMM suite. The URL is set at installation time and normally should not be changed. 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 name of the computer cannot be `localhost`.

### Default value

Not defined

### Example

The following example shows the server URL in an environment that is configured for SSL:

`https://machineName.companyDomain.com:8080/customer/unica`

## Display Name

### Description

The **Display Name** property is used internally. Do not change the value.

### Default value

`InteractionHistory`

## Scheduler Edit Page URI

### Description

The **Scheduler Edit Page URI** property is used internally. Do not change the value.

### Default value

`jsp/scheduleOverride.jsp?taskId=`

## Logout URL

### Description

The **Logout URL** property is used internally. Do not change the value.

### Default value

`logout.do`

## Interaction History | partitions | partition[n] | datasource

The properties that are listed in the **Interaction History | partitions | partition[n] | datasource** category specify details about the Interaction History system tables. For multiple partitions, each partition includes this subcategory.

### Database Type

#### Description

The **Database Type** property specifies the database type of the Interaction History system tables for the partition.

#### Default value

DB2

#### Valid values

DB2

### JNDI Name

#### Description

The **JNDI Name** property specifies the JNDI name that is used in the web application server for the JDBC connection to the Interaction History system tables.

#### Default value

[Change me]

### Interaction History DSN

#### Description

The **Interaction History DSN** property specifies the name of the Interaction History data source.

For DB2 database, set the value of the property to the DB2 instance name.

#### Default value

[Change me]

### Datasource UserName

#### Description

The **Datasource UserName** property must be set to the login name of the IBM EMM user account that has the data source that holds the database credentials for the Interaction History system table database or schema.

#### Default value

[Change me]

### Interaction History DSN Database (Only for DB2)

#### Description

The **Interaction History DSN Database (Only for DB2)** property specifies the name of the database or schema that holds the Interaction History system tables.

#### Default value

[Change me]

## Interaction History | partitions | partition[n] | configuration

The properties that are listed in the **Interaction History | partitions | partition[n] | configuration** category specify the minimum value for a response that must be included in reports. You can also specify the data of records that must be imported from Campaign, Interact, and eMessage.

### ThresholdValueForResponse

#### Description

Reports should not show stimuli that are associated with a small amount of credit. The **ThresholdValueForResponse** property eliminates the smaller credits so that they do not overwhelm the report.

Any contact that receives credit for a response with a value lower than the threshold set by this property is not included in reports. Instead, this credit is distributed to other contacts that are eligible to receive credit for this response.

#### Default value

0.05

## Start date for initial ETL (MM-DD-YYYY)

#### Description

The **Start date for initial ETL (MM-DD-YYYY)** property sets the earliest date of records to be imported. It applies only to the first import of data from Campaign, Interact, and eMessage.

Enter a date value in MM-DD-YYYY format.

When the value of the property is not set, or is a future date, or is an incorrect format, the system defaults to a date 90 days in the past.

#### Default value

[Change me]

## Interaction History | partitions | partition[n] | CoreMetrics

The properties in the **Interaction History | partitions | partition[n] | CoreMetrics** category specify details of the data that IBM Digital Analytics uploads for each partition in Interaction History.

For each partition, properties in the **Interaction History | partitions | partition[n] | CoreMetrics** category specify the following details:

- The FTP server where IBM Digital Analytics uploads data that is exported for use by Interaction History.
- The IBM Digital Analytics data that is exported for use by Interaction History.

### FTP Root Directory

#### Description

The **FTP Root Directory** property specifies the directory on the FTP server where IBM Digital Analytics uploads data that is exported for use by Interaction History.

**Default value**

[Change me]

**FTP Server****Description**

The **FTP Server** property specifies the name or IP address of the FTP server where IBM Digital Analytics uploads data that is exported for use by Interaction History.

**Default value**

[Change me]

**FTP Port****Description**

The **FTP Port** property specifies the port on which the FTP server listens.

**Default value**

21

**Datasource UserName****Description**

The **Datasource UserName** property must be set to the login name of the IBM EMM user account that has the data source that holds the credentials for the FTP server where IBM Digital Analytics uploads data that is exported for use by Interaction History.

**Default value**

[Change me]

**CoreMetrics ClientID****Description**

Set the value of the **CoreMetrics ClientID** property to the unique IBM Digital Analytics Client ID assigned to your company.

**Default value**

[Change me]

**Feed Name****Description**

The **Feed Name** property is used internally. Do not change the value.

**Staging Directory****Description**

Set the value of the **Staging Directory** property to the name of a directory on the computer where Interaction History is installed. IBM Digital Analytics data feed is stored in this directory temporarily during data imports.

**Default value**

[Change me]

## Cost of Contact

### Description

The **Cost of Contact** property specifies the cost of each contact that is recorded in IBM Digital Analytics.

### Default value

0

## Cost of Response

### Description

The **Cost of Response** property specifies the cost of each response that is recorded in IBM Digital Analytics.

### Default value

0

## Default Channel

### Description

The **Default Channel** property specifies the name that is given to the web channel in your Interaction History reports. You must give the web channel the same name when you map channels on the Interaction History Settings page.

### Default value

[Change me]

## Audience Level Name

### Description

The **Audience Level Name** property specifies the name that is used in Campaign for the audience level from which contact and response history are used. Only one audience level can be used.

### Default value

[Change me]

## Default Cell Name

### Description

The **Default Cell Name** property specifies the name that you want to assign to the segment that contains the IBM Digital Analytics responses. The name is used in Interaction History reports that filter data by segment.

### Default value

[Change me]

## Source of Audience Mapping

### Description

The **Source of Audience Mapping** property indicates whether the translation table is a flat file or a database table.

**Default value**

File

**Valid values**

File | Table

## Data Source for Translation Table

**Description**

The **Data Source for Translation Table** property specifies the name of the JDBC data source that connects to the translation table. This JDBC data source is created in the web application server where Interaction History is deployed.

The property is used only when the **AudienceIDMappingSrc** property is set to Table.

**Default value**

[Change me]

## Translation Table Name

**Description**

The **Translation Table Name** property specifies the name of the translation table that is used to translate the IBM Digital Analytics keys to Campaign audience keys.

The property is used only when the **AudienceIDMappingSrc** property is set to Table.

**Default value**

[Change me]

## Translation Table Auto Incremental Field

**Description**

The **Translation Table Auto Incremental Field** property specifies the name of the column in the translation table that is of the auto increment number type. Interaction History uses this column to determine which are the new records added in this table.

The first time the import is performed, Interaction History imports all the available data. When the **Translation Table Auto Incremental Field** property is set, Interaction History imports only the new contacts in subsequent imports. If the column is not specified, then all the records are imported every time; this might cause slow performance.

The property is used only when the **AudienceIDMappingSrc** property is set to Table.

**Default value**

Not defined

## CMRegIdColumn

**Description**

The **CMRegIdColumn** specifies the name of the column in the translation table that holds the IBM Digital Analytics registration ID.

The property is used only when the **AudienceIDMappingSrc** property is set to Table.

#### Default value

Not defined

## CampaignColumn[n]

#### Description

There are five **CampaignColumn[n]** properties (**CampaignColumn1**, **CampaignColumn2**, and so on). They have corresponding properties for the equivalent columns in the translation table (**TTColumn1**, **TTColumn2**, and so on).

The property is used only when the **AudienceIDMappingSrc** property is set to Table.

Use the following guidelines to set the **CampaignColumn[n]** and **TTColumn[n]** properties:

- If your audience level in Campaign is not a compound audience level, set **CampaignColumn1** to the name of the database column that holds the Campaign audience level. Set **TTColumn1** to the name of the column in your translation table that holds the Campaign audience level.
- If your audience level is a compound audience level that is composed of multiple columns, use as many of the **CampaignColumn** and **TTColumn** properties as necessary, one pair for each part of your audience level.

For example, suppose that you have a compound audience level in Campaign, composed of two columns, custid and emailid. The following table shows the portion of your translation table that holds the Campaign audience level parts:

*Table 13. Example mapping in translation table*

Translation table column	Campaign audience column
CampAud1	custid
CampAud2	emailid

Set the following configuration properties:

- **CampaignColumn1**: custid
- **TTColumn1**: CampAud1
- **CampaignColumn2**: emailid
- **TTColumn2**: CampAud2

#### Default value

Not defined

## TTColumn[n]

#### Description

There are five **TTColumn[n]** properties (**TTColumn1**, **TTColumn2**, and so on).

The property is used only when the **AudienceIDMappingSrc** property is set to Table.

**Default value**

Not defined

## Interaction History | partitions | partition[n] | CampaignAndInteract

The properties in the **Interaction History | partitions | partition[n] | CampaignAndInteract** category specify the Campaign and Interact data source in a partition. Each partition that you add includes this sub category.

### Database Type

**Description**

The **Database Type** property specifies the database type of the Campaign system tables in this partition.

**Default value**

DB2

**Valid values**

DB2

### Campaign DSN

**Description**

The **Campaign DSN** property specifies the name of the Campaign data source.

For DB2 database, set the value of the property to the DB2 instance name.

**Default value**

[Change me]

### Datasource User Name

**Description**

The **Datasource User Name** property specifies the login name of the IBM EMM user account that has the data source that holds the database credentials for the Campaign system table database or schema.

**Default value**

[Change me]

### Campaign DSN Database (Only for DB2)

**Description**

The **Campaign DSN Database (Only for DB2)** property specifies the name of the database or schema that holds the Campaign system tables.

**Default value**

[Change me]

## Interaction History | partitions | partition[n] | eMessage

The properties in the **Interaction History | partitions | partition[n] | eMessage** category specify the eMessage data source in a partition.

The values that you set for the database type, DSN, and login properties in this category are most often the same as those you set for the equivalent properties in the **Interaction History | partitions | partition[n] | CampaignAndInteract** category.

Each partition that you add includes this sub category.

## **Default Channel**

### **Description**

The **Default Channel** property specifies the name that is given to the email channel in your Interaction History reports. You must give the email channel the same name when you map channels on the Interaction History Settings page.

### **Default value**

Email

## **eMessage Contact Cost**

### **Description**

The **eMessage Contact Cost** property specifies the cost of each email contact in the partition.

### **Default value**

0

## **eMessage Response Cost**

### **Description**

The **eMessage Response Cost** property specifies the cost of each email response in the partition.

### **Default value**

0

## **eMessage TreatmentCode Prefix**

### **Description**

The **eMessage TreatmentCode Prefix** property specifies the name of the parameter that is used in eMessage to hold the treatment code.

### **Default value**

emsg

## **Interaction History | partitions | partition[n] | Reports**

The property in the **Interaction History | partitions | partition[n] | Reports** category specifies the name of the folder in Cognos used for reports. The name of the folder is different for each partition.

## **Analysis\_Report\_Folder**

### **Description**

The **Analysis\_Report\_Folder** property specifies the name of the folder in Cognos used for reports. This is different for each partition in your system.

Retain the syntax that is shown in the default value, but change the value that is shown as Interaction History if necessary for each partition.

#### Default value

```
/content/folder[@name='Interaction History']
```

---

## Attribution Modeler configuration properties

Log in to the IBM Enterprise Marketing Management Suite, and then browse to **Settings > Users** to set the Attribution Modeler configuration properties on the Configuration page. You can use all the features of Attribution Modeler after you set the configuration properties.

### Attribution Modeler | navigation

The properties that are listed in the **Attribution Modeler | navigation** category specify values that are used internally to navigate among IBM products. All the properties apply across all partitions.

#### httpPort

##### Description

The **httpPort** specifies the HTTP port in the web application server on which Attribution Modeler is deployed. The property is used internally for communication among IBM products.

##### Default value

7001

#### httpsPort

##### Description

The **httpsPort** specifies the SSL port in the web application server on which Attribution Modeler is deployed. The property is used internally for communication among IBM products, when SSL is enabled.

##### Default value

7001

#### serverURL

##### Description

The **serverURL** specifies the URL that is used for the IBM EMM suite. The URL is set at installation time and normally should not be changed. 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 name of the computer cannot be localhost.

##### Default value

`http://localhost:7001/am`

##### Example

The following example shows the server URL in an environment that is configured for SSL:

`https://machineName.companyDomain.com:8080/am`

## **logoutURL**

### **Description**

The **logoutURL** property is used internally. Do not change the value.

### **Default value**

logout.do

## **displayName**

### **Description**

The **displayName** property is used internally. Do not change the value.

### **Default value**

Attribution Modeler

## **AttributionModeler | AMListener**

Properties in this category specify values that are used internally and that apply across all partitions.

## **serverHost**

### **Description**

Set to the name or IP address of the machine on which the Attribution Modeler listener is installed.

### **Default value**

localhost

## **logStringEncoding**

### **Description**

The encoding that is used for the Attribution Modeler listener log.

This value should match the encoding that is 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 the *IBM Campaign Administrator's Guide* for a list of supported encodings.

## **systemStringEncoding**

### **Description**

This property specifies the encoding that is used to interpret values that come into Attribution Modeler from the operating system (file system paths and file names, for example), and the encoding in which Attribution

Modeler presents values back to the operating system. This value is generally set to native. For multi-locale environments, UTF-8 is the preferred setting.

The value can include more than one encoding, which is 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 the *IBM Campaign Administrator's Guide* for a list of supported encodings.

## **loggingLevel**

**Description**

This property determines how much information is recorded in the listener log. Setting the logging level to HIGH or ALL might affect performance.

**Default value**

MEDIUM

**Valid values**

LOW, MEDIUM, HIGH, ALL

## **serverPort**

**Description**

The value of this property sets the port on which the Attribution Modeler server listens.

**Default value**

5664

## **AttributionModeler | partitions | partition[n] | AMFields**

### **numFields**

**Description**

**Default value**

1

### **FieldName**

**Description**

**Default value**

Customer

**type**  
**Description**  
**Default value**  
    numeric

**name**  
**Description**  
**Default value**  
    OfferID

## Attribution Modeler | partitions | partition[n]

The properties in the **Attribution Modeler | partitions | partition[n]** category specify values that affect how Attribution Modeler evaluates data. There is one set of these properties for each partition.

### enableAttributionModeler

**Description**

The **enableAttributionModeler** property indicates whether the response attribution methods in Attribution Modeler is applied to your reports. Setting this property to False disables the SIRA, first touch, and equal credit attribution methods. Direct and Last touch attribution methods are still enabled to support reports that are a part of Interaction History.

Even though you can change this property value from true to false, it is expected that you set this value only one time in the beginning and never change it. If you want to change this property value after the first run, delete the IH\_AMTracking table beforehand to prevent some of the SIRA credits from being left unchanged. Then, rerun scoring in the new enabled or disabled state.

**Default value**

True

### enableBackgroundCredit

**Description**

The **enableBackgroundCredit** property is used to represent all the unknown precursor events of a response with a single “background” offer and compute a proportion of the credit for it. Instead of ignoring the cases where a customer response does not have an earlier contact, the background credit is used to represent all of the previously unknown events of a response. A false contact is created for each customer automatically whenever a customer has at least one response, whether they have any contacts or not.

Even though you can change this property value from true to false, it is expected that you set this property only one time in the beginning and never change it. If you do change the property value between runs, clear the tracking table before changing this property to prevent scoring issues.

This property is enabled only if **enableAttributionModeler** is set to True.

**Default value**

True

**Valid values**

True | False

## Training Period

**Description**

The **Training Period** property sets the number of days, looking backward from the current date, for which contacts and responses are pulled from Interaction History to train the Attribution Modeler model during each run.

Set the property so that you obtain a representative sample, which is based on your business model, how frequently your offers change, and your marketing goals.

**Default value**

40

**Valid values**

Any integer

## Attribution Modeler | partitions | partition[n] | dataSources

This category is empty when you first install Attribution Modeler. For each partition in your system, you must import configuration properties that enable you to specify many details of how Attribution Modeler interacts with its system tables.

You import a set of configuration properties appropriate for the type of database or schema that holds your system tables.

This procedure is described in the *Interaction History and Attribution Modeler Installation Guide*.

The properties in this category are identical to the properties of the same name that are used in Campaign.

For details about the properties, see the *IBM Marketing Platform Administrator's Guide*, which contains descriptions of configuration properties for all the Enterprise products.

If you are using context help on the Configuration page, click **Help > Help for this page**, and on the first landing page that opens, click the link for Campaign. Then, continue to follow links to navigate to the **Campaign | Partitions | partition[n] | dataSources** category. The properties that are listed on the page that opens include the ones that are listed in the **Attribution Modeler | partitions | partition[n] | dataSources** category.

## AttributionModeler | partitions | partition[n] | server | encoding

These configuration properties are for encoding settings.

There is one of these properties for each partition.

### stringEncoding

**Description**

This property specifies how Attribution Modeler reads in and writes out flat files. It should match the encoding that is 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 Attribution Modeler.

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 can always explicitly set the `StringEncoding` property in the `dataSources` category.

See the *IBM Campaign Administrator's Guide* for a list of supported encodings.

#### Default value

No default value defined.

## AttributionModeler | partitions | partition[n] | server | logging

Properties in this category specify values that affect how Attribution Modeler handles logging. There is one set of these properties for each partition.

### loggingCategories

#### Description

The `loggingCategories` property specifies the category of messages that are written to the Attribution Modeler server log file. This works with the `loggingLevels` property, which determines which messages are logged based on severity (for all selected categories). You can specify multiple categories in a comma-separated list. The special category `all` provides a shorthand for specifying all logging categories.

#### Default value

ALL

#### Valid values

Supported categories are:

- 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

## **loggingLevel**

### **Description**

This property controls the amount of detail that is written to the Attribution Modeler server log file, `AMsvr.log`, based on severity.

### **Default value**

MEDIUM

### **Valid values**

- LOW
- MEDIUM
- HIGH
- ALL

LOW represents the least detail (the most severe errors only), and ALL includes trace messages and is intended primarily for diagnostic purposes.

**Note:** You may want to set the `loggingLevels` property to ALL during configuration and testing to maximize the logging output from Attribution Modeler for diagnostic purposes. This setting generates a large amount of data, and, therefore, may not be advisable for production operation.

## **logMaxFileSize**

### **Description**

This property specifies the maximum size, in bytes, that the Attribution Modeler server log file is allowed to reach before it is rolled over to back up files.

### **Default value**

10485760 (10 MB)

## **logMaxBackupIndex**

### **Description**

This property specifies the number of backup Attribution Modeler 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 that is 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)

## **enableLogging**

### Description

This property turns on or off logging for the partition that is specified in the partitions category under which this property is found.

### Default value

TRUE

## **Attribution Modeler | partitions | partition[n] | AdvancedOptions**

The properties in the **Attribution Modeler | partitions | partition[n] | AdvancedOptions** category specify values that affect how Attribution Modeler evaluates data, and that apply across all partitions.

## **sampleSize**

### Description

The **sampleSize** property defines the percentage of available records that are used for training. This value needs to be set to a number that is greater than 0 but less than 100 (percent).

### Default value

100

## **randomSeed**

### Description

The **randomSeed** property represents the starting point that Attribution Modeler uses to select records randomly.

### Default value

123456789

## **maxTrainingTime**

### Description

The **maxTrainingTime** property specifies the maximum time, in minutes, that Attribution Modeler spends training itself. It sets a time limit on the training process as it iterates over the data to reach the goal that is set by the **convergenceThreshold** property. This time limit helps you limit the resources that Attribution Modeler uses. A warning is logged in the log file if SIRA exceeds this training time limit.

### Default value

60

## **convergenceThreshold**

### Description

The **convergenceThreshold** property is used to set a limit for how much difference is allowed between the results of one training iteration and the

next. This difference is expressed as a percentage of responses for which the results (winning offers) are allowed to change from one iteration to the next.

If you set the property to 0 (zero), then you are not allowing any changes to the results from one training iteration to the next; this is the most rigorous standard. If you set this property to a value higher than 0, then you are allowing the training results to be more flexible; the standard is less rigorous and training might be completed sooner.

**Default value**

3

## **noiseEliminationThreshold**

**Description**

The **noiseEliminationThreshold** property is used to eliminate the SIRA credits below the threshold value.

**Default value**

3

---

## **Report configuration properties**

The report configuration properties for IBM EMM are located at **Settings > Configuration > Reports**.

To generate reports, the IBM EMM suite integrates with IBM Cognos, a third-party business intelligence application. You use the **Integrations > Cognos** properties to identify your IBM Cognos system. Then, for Campaign, eMessage, and Interact, there are additional configuration properties to set up and customize the reporting schemas.

### **Reports | Integrations | Cognos [version]**

This page displays properties that specify URLs and other parameters for the IBM Cognos system used by this IBM system.

#### **Integration Name**

**Description**

Read-only. Specifies that IBM Cognos is the third-party reporting or analytical tool used by the IBM EMM 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 specified by the Integration Name property.

**Default value**

Cognos

## **Version**

### **Description**

Read-only. Specifies the product version of the application 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 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`

Note that 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 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. Note that this user also has access to IBM EMM.

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 is true:

- When the IBM EMM is not installed in the same domain as the IBM Cognos applications.
- When IBM Cognos is accessed using an IP address (within the same network domain) instead of the Fully Qualified Hostname (which is being used to access the IBM EMM applications), even if both the IBM EMM 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 EMM 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 should install IBM Cognos and IBM EMM in the same domain.

**Default value**

False

**Valid Values**

True | False

## **Reports | Schemas | [product] | [schema name] | SQL Configuration**

### **Table/View Name**

**Description**

Specifies the name of the view or table that the SQL script you generate for this reporting schema will create. 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 UPPER-CASE letters

Following is the naming convention you should use:

- Start the name with the letter "UAR"
- Add a one-letter code to represent the IBM EMM 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 list of codes, below.

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 EMM 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**

### **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 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 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 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 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 supported by this reporting schema.

**Default value**

`UA_ResponseHistory`

## **Over Time Variations**

### **Description**

Specifies the calendar time periods 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]**

Use this form 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 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 UPPER-CASE 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 | [schema name] | Columns | [Response Metric]

Use this form to add the response metrics 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 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 UPPER-CASE 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 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 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 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 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 supported by this reporting schema.

**Default value**

UA\_ResponseHistory

## **Over Time Variations**

**Description**

Specifies the calendar time periods 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**

This schema supports reporting on campaign-detailed responses, 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.

### **Response History Table**

**Description**

Specifies the name of the Response History table for the audience level supported by this reporting schema.

**Default value**

UA\_ResponseHistory

## **Reports | Schemas | Campaign | Campaign Offer Response Breakout | Columns | [Response Type]**

Use this form to add to the reporting schema any custom response types you want to include in your reports.

### **Column Name**

**Description**

Specifies the name to use in the reporting view or table for the column 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 UPPER-CASE letters, and it cannot have spaces.

### Response Type Code

#### Description

The response type code for the specified response type. This is the value 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 EMM 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 | Campaign Offer Contact Status Breakout

This schema supports reporting on campaign-detailed contacts, 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 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 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 supported by this reporting schema.

#### Default value

UA\_Dt1ContactHist

## Reports | Schemas | Campaign | Campaign Offer Contact Status Breakout | Columns | [Contact Status]

### Column Name

#### Description

Specifies the name to use in the reporting view or table for the column 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 UPPER-CASE letters, and it cannot have spaces.

## Contact Status Code

### Description

The name of the contact status code. This is the value 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 | [Campaign Custom Column]

Use this form to add to the reporting schema any custom campaign attributes that you want to include in your reports.

### Column Name

### Description

Specifies the name to use in the reporting view or table for the attribute 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 UPPER-CASE 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 | Campaign Custom Attributes | Columns | [Offer Custom Column]

Use this form to add to the reporting schema any custom offer attributes that you want to include in your reports.

### Column Name

#### Description

Specifies the name to use in the reporting view or table for the attribute 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 UPPER-CASE 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 this form to add to the reporting schema any custom cell attributes that you want to include in your reports.

### **Column Name**

#### **Description**

Specifies the name to use in the reporting view or table for the attribute 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 UPPER-CASE 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, run time, and learning databases. Use the properties from this page to specify the JNDI names of the data sources for those databases.

The data sources 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.

Note that the database type of the data sources must match the database type 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 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 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 supported by this reporting schema.

### **Default value**

UA\_ResponseHistory

## **Over Time Variations**

### **Description**

Specifies the calendar time periods 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**

### **eMessage Tracking Datasource (JNDI)**

### **Description**

Specifies the name of the JNDI data source that identifies the eMessage tracking tables, which are located 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 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 supportedLocales**

### **Description**

Specifies the locales available in your installation of IBM Marketing Operations. List only the locales that you are actually using. Each locale you list uses memory on the server. The amount of memory 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**

**welcomePageURI**

**Description**

The Uniform Resource Identifier of the IBM Marketing Operations index page. This value is used internally by IBM EMM 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 EMM applications. Changes to this value are not recommended.

**Default value**

blank

**seedName**

**Description**

Used internally by IBM EMM applications. Changes to this value are not recommended.

**Default value**

Plan

**type**

**Description**

Used internally by IBM EMM 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.

#### **Default value**

`http://<server>:<port>/plan`

### **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 **Marketing Operations > about** configuration properties 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](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**

### **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 visited 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 creating 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 visit (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 visit.

### **Default value**

True

## **enableRevisionHistoryPrompt**

### **Description**

Ensures that users are prompted to add change comments when saving a project/request or approval.

### **Valid values**

True | False

### **Default value**

False

## **useForecastDatesInTaskCalendar**

### Description

Specifies the type of dates used when displaying tasks 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), and tab security is enabled for custom tabs.
- False: user access to projects and requests is evaluated only according 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 created from templates, including programs, plans, and invoices.

**Valid values**

- True enables unique internal IDs across all objects 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 created from templates.

**Default value**

True

## **FMEabled**

### **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

## **FMPProjVendorEnabled**

### **Description**

Parameter used to show/hide vendor column for project line items.

### **Valid values**

True | False

### **Default value**

False

## **FMPrgmVendorEnabled**

### **Description**

Parameter used to show/hide vendor column for program line items.

### **Valid values**

True | False

### **Default value**

False

## **Marketing Operations | umoConfiguration | 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

## Marketing Operations | umoConfiguration | templates

**Important:** Changes to the default values supplied for these parameters are not recommended.

### **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_EMM_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**

**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 | Email****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 have not provided IBM Marketing Operations with an existing JavaMail session that uses the session parameter above 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**

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 made previously in a web browser with the native Marketing Operations method.  
If you specify SOAP, you must also configure the **markupServerURL** and **useCustomMarkup** parameters.
- 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 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`

## **useCustomMarkup**

### **Description**

Determines whether Windows users can send and receive markup comments using the **Acrobat Send Receive Comments** button.

### **Valid values**

- True: Windows users can use only the **Acrobat Send Receive Comments** button to send and receive markup comments. The `UMO_Markup_Collaboration.js` file must be available in the `javascripts` folder of the client-side Acrobat installation.  
Dependent on **markupServerType** = SOAP.
- False: Windows users can use only the Marketing Operations custom **Send Comments** button to send and receive markup comments. They cannot use the Acrobat button and must configure Acrobat to enable the IBM Marketing OperationsComments toolbar. For more information about reviewing PDF files, see the *IBM Marketing Operations User's Guide*.

### **Default value**

True

## **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 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

## **griddatasd**

### **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****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 enableIntegrationServices**

**Description**

Enables and disables the Integration Services module that can be used by third-party users to access IBM Marketing Operations functionality using web services and triggers.

**Valid values**

True | False

**Default value**

False

**integrationProcedureDefinitionPath****Description**

Optional full file path to the custom procedure definition XML file.

**Default value**

*<IBM\_EMM\_Home>/<MarketingOperations\_Home>/devkits/integration/examples/src/procedure/procedure-plugins.xml*

**integrationProcedureClasspathURL****Description**

URL to the class path for custom procedures.

**Default value**

*file:///<IBM\_EMM\_Home>/<MarketingOperations\_Home>/devkits/integration/examples/classes/*

**Marketing Operations | umoConfiguration | campaignIntegration****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 | 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 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**

### **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 EMM applications.

#### **Default value**

blank

### **thresholdForUseOfSubSelects**

#### **Description**

Specifies the number of records beyond which a subquery should 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 lot of application data. 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 listItemsPerPage**

### **Description**

Specifies how many items (rows) are displayed in one list page. This value should be greater than 0.

### **Default value**

**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 (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  
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 trueTypeFontDir**

### **Description**

Specifies the directory where the True Type fonts are located. This parameter is required for thumbnail generation on non-Windows platforms that use Aspose. For Windows installations, this parameter is optional.

### **Default value**

blank

## **coreThreadPoolSize**

### **Description**

Specifies the number of persistent threads kept in the thread pool for thumbnail generator threads.

### **Default value**

5

## **maxThreadPoolSize**

### **Description**

Specifies the maximum number of threads 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**

**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 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 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 machine.

**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 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

**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 allowed by this parameter.

For example, **notifyEMailMonitorPollPeriod** is set to poll every 60 seconds. Setting this **notifyEMailMonitorMaximumResends** property to 60 attempts causes the email monitor to try to resend a failed message once in each poll (that is, every minute), for up to one 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**

### **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****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****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 that IBM 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**

### **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**

### **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**

### **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.SimpleDateFormat` 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**

### **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**

**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****Description**

The seedName property is used internally by IBM applications. You should not change this value.

**Default value**

No default value defined.

**type****Description**

The Campaign > navigation > type property is used internally by IBM applications. You should not change this value.

**Default value**

No default value defined.

**httpPort****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****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****Description**

The Campaign > navigation > 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`

**Default value**

`http://localhost:7001/Campaign`

**serverURLInternal****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 EMM 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**

#### **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**

#### **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=`

### **offerDetailPageURI**

#### **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**

#### **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=`

### **displayName**

#### **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 | caching

The properties in the caching category specify the length of time that cached data for channels, initiatives, campaigns, sessions, and offers is retained.

### **offerTemplateDataTTLSeconds**

#### 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**

#### 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**

#### 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**

#### 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**

#### 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**

### **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**

### **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**

### **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 I partitions**

This category contains properties used to configure all 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 you configure in Campaign.

### **Campaign I partitions I partition[n] I eMessage**

Properties in this category allow you to define characteristics of recipient lists and specify the location of resources that upload the lists to IBM EMM 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 you specify must be the full local directory path in the file system for the machine that hosts the Campaign web application server.

The IBM installer populates this setting automatically for the default partition when you run the installer. For additional 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 the machine 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 using 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 history 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] | reports

These configuration properties define 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 using 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 using 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 using 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 using 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 using 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 using 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 using 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 using 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 have installed 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 have installed 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**

The `partition[n] > audienceLevels` category contains sub-categories and properties that are created and populated when a user creates audience levels in Campaign. You should not edit properties in this category.

Properties in the `partition[n] > audienceLevels > audienceLevel` category specify the number of fields in the audience level and the name of an audience level. These properties are populated when a user creates audience levels udience levels in Campaign. You should not edit properties in this category.

### **numFields**

#### Description

This 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.

### **audienceName**

#### Description

This 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] | 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

Choose Campaign | Partitions | partition[n] | dataSources to configure how IBM Campaign interacts with databases, including its own system tables.

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 Campaign is represented by a category under partition[n] > dataSources > DATA\_SOURCE\_NAME.

**Note:** The Campaign system tables data source for each partition must be named UA\_SYSTEM\_TABLES in the Marketing Platform, and a dataSources > UA\_SYSTEM\_TABLES category must exist in the Configuration page for every Campaign partition.

### AccessLibrary

##### Description

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 Campaign environment. For example, 64-bit 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 Campaign to access, for example, Teradata).

### **Additional libraries for AIX®**

#### **Description**

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 as required (for example, to `libodb4dDD.so`, overriding the default selection of `libodb4d.so`).

#### **Default value**

No default value defined.

### **AliasPrefix**

#### **Description**

The `AliasPrefix` property specifies the way Campaign forms the alias name that 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

### **AllowBaseJoinsInSelect**

#### **Description**

This property determines whether 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**

#### **Description**

The `AllowSegmentUsingSQLCase` property specifies whether the Campaign 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, 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**

#### **Description**

The AllowTempTables property specifies whether 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.

When temporary tables are enabled, each time a query is issued against the database (for example, by the Segment process), the resulting IDs are written in a temporary table in the database. When an additional query is issued, Campaign may use that temporary table to retrieve rows from the database.

If temporary tables are not enabled, Campaign retains the selected IDs in the server memory. The additional query retrieves IDs from the database and matches them to the IDs in server memory.

For more information about controlling temporary table joins, see MaxTempTableJoinPctSelectAll and MaxTempTableJoinPctWithCondition.

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

### **ASMSaveDBAuthentication**

#### **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, Campaign saves your user name and password in IBM EMM.

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****Description**

The `ASMUserForDBCredentials` property specifies the IBM EMM user name that is assigned to the Campaign system user (required for accessing the Campaign system tables).

This property is undefined by default.

**Default value**

No default value defined.

**BulkInsertBlockSize****Description**

The `BulkInsertBlockSize` 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****Description**

The `BulkInsertRequiresColumnType` property is required to support Data Direct ODBC data sources only. Set this property to TRUE for Data Direct 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****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****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**

### **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**

### **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**

### **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 your database uses, select a date from the database as described below.

### **Selecting a date by database**

*Table 14. Date formats*

Database	To determine the correct setting
DB2	<p>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:</p> <pre>values current date</pre> <p>If your operating system does not provide the db2test utility, use the cxntest utility to test connections to the target database.</p>
Netezza®	<p>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:</p> <pre>CREATE TABLE date_test (f1 DATE); INSERT INTO date_test values (current_date); SELECT f1 FROM date_test;</pre> <p>Another way to select date format is to run following command:</p> <pre>SELECT current_date FROM ANY_TABLE limit 1;</pre> <p>where ANY_TABLE is the name of any existing table</p>
Oracle	<p>Log in to the database from the machine that is running the Campaign server. Use SQL *Plus to connect and issue the following command:</p> <pre>SELECT sysdate FROM dual</pre> <p>The current date is returned in NLS_DATE_FORMAT for that client.</p>
SQL Server	<p>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:</p> <pre>SELECT getdate()</pre>

### Additional considerations

Note the following database-specific instructions.

#### 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 our system table settings, use:

- SuffixOnCreateDateField = FORMAT 'YYYY-MM-DD'
- DateFormat = DELIM\_Y\_M\_D
- DateOutputFormatString = %Y-%m-%d

#### SQL Server

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.

#### Default value

DELIM\_Y\_M\_D

### **Valid Values**

Any of the formats that are specified in the DATE macro

### **DateOutputFormatString**

#### **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\_str in the DATE\_FORMAT macro. The DATE\_FORMAT macro accepts two different kinds of formats. One is an identifier (for example, DELIM\_M\_D\_Y, DDMMYYYY, 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**

#### **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 your database uses to display datetime/timestamp data on select. For most databases, this setting is the same as the setting for DateTimedOutputFormatString.

Typically, you set the DateTimeFormat by prepending your DateFormat value with DT\_ after determining the DateFormat value as described in the table for "Selecting a date by database".

**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

### **DateDateTimeOutputFormatString**

#### Description

The DateDateTimeOutputFormatString 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 DateDateTimeOutputFormatString 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] > DateFormat property.

See DateOutputFormatString for a method for verifying that the format you select is correct.

#### Default value

%Y/%m/%d %H:%M:%S

### **DB2NotLoggedInitially**

#### Description

The DB2NotLoggedInitially property determines whether Campaign uses the not logged initially SQL syntax when populating temporary tables in DB2. When set to TRUE, this property disables logging for inserts in to temp tables, which improves performance and decreases database resource consumption.

If your version of DB2 does not support the not logged initially syntax, set this property to FALSE.

#### Default value

TRUE

#### Valid Values

TRUE | FALSE

### **DB2NotLoggedInitiallyUserTables**

#### Description

The DB2NotLoggedInitiallyUserTables property determines whether Campaign uses the not logged initially SQL syntax for inserts into DB2 user tables. When set to TRUE, this property disables logging for inserts into the user tables, which improves performance and decreases database resource consumption.

**Note:** 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.

**Note:** The DB2NotLoggedInitiallyUserTables property is not used for the Campaign system tables.

#### Default value

FALSE

#### Valid Values

TRUE | FALSE

### DefaultScale

#### Description

The DefaultScale property is used when Campaign creates a database field to store numeric values from a flat file, 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.)

Flat files do not indicate precision and scale. Use DefaultScale to specify how many places to the right of the decimal point to define for the field that is created. For example:

- 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

#### 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**

**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**

**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**

**Description**

The DisallowTempTableDirectCreate 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****Description**

Set this property to the data source name (DSN) as assigned in your ODBC configuration for this Campaign data source. 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 would have a different name in Campaign, but if they refer to the same physical data source and they will have the same DSN value.

**Default value**

No default value defined.

**DSNUseOSAuthentication****Description**

The DSNUseOSAuthentication 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****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****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**

#### **Description**

The `EnableSelectOrderBy` property specifies whether the internal lists of IDs for Campaign are 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**

#### **Description**

The `ExcludeFromTableDisplay` parameter allows you to limit the database tables that are displayed during table mapping in Campaign. It does not reduce the number of table names retrieved from the database.

Table names matching the specified patterns are not displayed.

For example, if you set the value of this parameter to `sys.*`, tables with names that begin with `sys.` are not displayed. Note that the values for this parameter are case-sensitive.

#### Default value

`UAC_*`, which excludes temp tables and Extract tables, when the `ExtractTablePrefix` property's value is the default value

### ExtractTablePostExecutionSQL

#### 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.

*Table 15. Tokens available to ExtractTablePostExecutionSQL*

Token	Description
<code>&lt;AMUSER&gt;</code>	This token is replaced with the IBM EMM user name associated with the flowchart for which the Extract table was created.
<code>&lt;CAMPAIGNCODE&gt;</code>	This token is replaced with the code for the campaign associated with the flowchart for which the Extract table was created.
<code>&lt;CAMPAIGNNAME&gt;</code>	This token is replaced with the name of the campaign associated with the flowchart for which the Extract table was created.
<code>&lt;DBUSER&gt;</code>	This token is replaced with the database user name for the database where the Extract table was created.
<code>&lt;FLOWCHARTNAME&gt;</code>	This token is replaced with the name of the flowchart associated with the Extract table creation.
<code>&lt;KEYCOLUMNS&gt;</code>	This token is replaced with the Extract table column name(s).
<code>&lt;TABLENAME&gt;</code>	This token is replaced with the Extract table name.
<code>&lt;USER&gt;</code>	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

#### Description

The `ExtractTablePrefix` property specifies a string that is automatically prepended to all Extract table names in Campaign.

**Default value**

UAC\_EX

**ForceNumeric****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

**InactiveConnectionTimeout****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****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****Description**

The JndiName property is used only when configuring the Campaign system tables (not for other data sources, such as customer tables). Set its value to the Java Naming and Directory Interface (JNDI) data source that is defined in the application server (WebSphere or WebLogic).

**Default value**

campaignPartition1DS

## **LoaderCommand**

### **Description**

The LoaderCommand property specifies the command issued to invoke your database load utility in Campaign. If you set this parameter, 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 parameter also invokes the database loader utility mode when 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.

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. 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 parameter is undefined by default.

Tokens available to LoaderCommand are described below.

*Table 16. Tokens available to LoaderCommand*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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 Campaign generates according to the template that is specified in the LoaderControlFileTemplate parameter.
<DATABASE>	This token is replaced with the name of the data source that 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 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.
<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.

*Table 16. Tokens available to LoaderCommand (continued)*

Token	Description
<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, but is supported for compatibility with earlier versions. See <TABLENAME>, which replaced <TABLE> as of version 4.6.3.
<TABLENAME>	This token is replaced with the database table name that Campaign is loading data into. This is the target table from your Snapshot process or the name of the Temp Table being created by 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

#### Description

The LoaderCommandForAppend parameter specifies the command issued to invoke your database load utility for appending records to a database table in Campaign. If you set this parameter, Campaign enters database loader utility mode for all output files from the Snapshot process that are used with the “append records” settings.

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 below.

*Table 17. Tokens available to LoaderCommandForAppend*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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.

*Table 17. Tokens available to LoaderCommandForAppend (continued)*

Token	Description
<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 parameter.
<DATABASE>	This token is replaced with the name of the data source that 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 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, but is supported for compatibility with earlier versions. See <TABLENAME>, which replaced <TABLE> as of version 4.6.3.
<TABLENAME>	This token is replaced with the database table name that Campaign is loading data into. This is the target table from your Snapshot process or the name of the Temp Table being created by 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.

#### LoaderControlFileTemplate

##### Description

The LoaderControlFileTemplate property specifies the full path and file name to the control file template configured in Campaign. When this parameter is set, Campaign dynamically builds a temporary control file based on the template that you specify here. The path and name of this temporary control file is available to the <CONTROLFILE> token that is available to the LoaderCommand parameter.

Before you use 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 Campaign.

For the correct syntax required for your control file, see your database loader utility documentation.

This parameter 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.

*Table 18. Tokens available to LoaderControlFileTemplate*

Token	Description
<code>&lt;DBCOLUMNNUMBER&gt;</code>	This token is replaced with the column ordinal in the database.
<code>&lt;FIELDLENGTH&gt;</code>	This token is replaced with the length of the field being loaded into the database.
<code>&lt;FIELDNAME&gt;</code>	This token is replaced with the name of the field being loaded into the database.
<code>&lt;FIELDNUMBER&gt;</code>	This token is replaced with the number of the field being loaded into the database.
<code>&lt;FIELDTYPE&gt;</code>	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 <code>&lt;FIELDLENGTH&gt;</code> token. For example, for SQLSVR and SQL2000 you would use "SQLCHAR(<FIELDLENGTH>)"
<code>&lt;NATIVETYPE&gt;</code>	This token is replaced with the actual database type that this field is loaded into.
<code>&lt;xyz&gt;</code>	This token places the specified character(s) on all fields being loaded into the database, except the last. A typical use is <code>&lt;,&gt;</code> which repeats a comma for all fields except the last.
<code>&lt;~xyz&gt;</code>	This token places the specified characters only on the last repeated line.
<code>&lt;!xyz&gt;</code>	This token places the specified character(s), including the angle brackets <code>&lt;&gt;</code> , on all lines.

#### Default value

No default value defined.

### **LoaderControlFileTemplateForAppend**

#### Description

The `LoaderControlFileTemplateForAppend` property specifies the full path and file name to the control file template configured in Campaign. When this parameter is set, Campaign 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 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 Campaign.

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.

### **LoaderDelimiter**

#### **Description**

The `LoaderDelimiter` property specifies whether the temporary data file is a fixed-width or delimited flat file, and, if it is delimited, the characters Campaign uses as delimiters.

If the value is undefined, 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. 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 can be enclosed in double quotation marks, if wanted.

### **LoaderDelimiterAtEnd**

#### **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, Campaign uses delimiters at the end of each line.

`FALSE`

#### **Default value**

`FALSE`

#### **Valid Values**

`TRUE` | `FALSE`

### **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, Campaign uses delimiters at the end of each line.

**Default value**

FALSE

**Valid Values**

TRUE | FALSE

**LoaderDelimiterForAppend**

**Description**

The LoaderDelimiterForAppend property specifies whether the temporary 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, 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. 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.

**LoaderUseLocaleDP**

**Description**

The LoaderUseLocaleDP property specifies, when 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**

**Description**

Allows you to specify the maximum number of items that 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

#### Description

The MaxQueryThreads property specifies the upper limit on the number of simultaneous queries allowed to run against each database source from a single Campaign flowchart.

Campaign runs database queries using independent threads. Because 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 value specified by this property, the Campaign server automatically limits the number of simultaneous queries to this value.

The maximum value is unlimited. Note that when the maxReuseThreads property 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

#### Description

When the selected number of IDs is less than the value specified by the MaxRowFetchRecords property, Campaign passes the IDs to the database, one at a time in a separate SQL query. This process may be very time-consuming. If the number of selected IDs is greater than the value specified by this parameter, Campaign uses temporary tables (if allowed against the database source), or it pulls down all the values from the table, not including any unnecessary values.

For performance reasons, it is best to keep this number low.

#### Default value

100

### MaxTempTableJoinPctSelectAll

#### Description

When a query is issued, 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, 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

#### Description

When a query is issued, 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

#### Description

Use this property to set the threshold for using the bulk loader. Campaign invokes the script assigned to the LoaderCommand 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, 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, Campaign assumes that the value is zero.

#### Default value

0 (zero)

#### Valid Values

Integers

### MinReqForLoaderCommandForAppend

#### Description

Use this property to set the threshold for using the bulk loader. 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, 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, Campaign assumes that the value is zero.

#### Default value

0 (zero)

#### Valid Values

Positive integers

## NumberOfRetries

### Description

The NumberOfRetries property specifies the number of times Campaign automatically retries a database operation on failure. Campaign automatically resubmits queries to the database this number of times before reporting a database error or failure.

### Default value

0 (zero)

## ODBCTableTypes

### Description

This property is empty by default, which is appropriate for all currently supported data sources.

### Default value

Not defined

### Valid Values

(empty)

## ODBCUnicode

### Description

The ODBCUnicode property specifies the type of encoding used in 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 - Campaign uses ANSI ODBC calls.
- UTF-8 - Campaign uses Unicode ODBC calls and assumes that a SQLWCHAR is a single byte. This is compatible with DataDirect ODBC drivers.
- UCS-2 - Campaign uses Unicode ODBC calls and assumes that a SQLWCHAR is 2 bytes. This is compatible with Windows and unixODBC ODBC drivers.

## ODBCv2

### Description

Use the ODBCv2 property to specify which ODBC API specification Campaign should use for the data source.

The default value of FALSE allows Campaign to use the v3 API specification, while a setting of TRUE causes 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, 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

#### Description

The OwnerForTableDisplay property allows you to limit the table mapping display in Campaign to tables owned by a specified user, or to one or more sets of tables owned by the specified user(s).

To display only those tables owned by one or more users, specify the database user IDs using a comma-separated list. For example:

```
<property name="OwnerForTableDisplay">user1,user2,user3</property>
```

To specify a table name pattern in addition to the user name, append the pattern to the user ID. For example, the following setting limits the table display to tables beginning with ABC for user1 and XYZ for user2:

```
OwnerForTableDisplay=user1.ABC%,user2.XYZ%
```

#### Default value

No default value defined.

### PadTextWithSpaces

#### Description

When set to TRUE, the PadTextWithSpaces property causes 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

#### Description

Use the PostExtractTableCreateRunScript property to specify a script or executable for Campaign to run after an Extract table has been created and populated.

Tokens available to PostExtractTableCreateRunScript are described below.

*Table 19. Tokens available to PostExtractTableCreateRunScript*

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 EMM 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**

#### Description

Specifies a script or executable that Campaign runs after a Segment temp table has been created and populated.

Tokens available to PostSegmentTableCreateRunScript are described below.

*Table 20. Tokens available to PostSegmentTableCreateRunScript*

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 EMM 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.

*Table 20. Tokens available to PostSegmentTableCreateRunScript (continued)*

Token	Description
<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**

#### Description

Use the PostSnapshotTableCreateRunScript property to specify a script or executable that Campaign runs after a Snapshot table has been created and populated.

Tokens available to PostSnapshotTableCreateRunScript are described below.

*Table 21. Tokens available to PostSnapshotTableCreateRunScript*

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 EMM 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**

#### 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.

*Table 22. Tokens available to PostTempTableCreateRunScript*

Token	Description
<DBUSER>	This token is replaced with the database user name for the database where the temp tables were created.
<AMUSER>	This token is replaced with the IBM EMM 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

#### 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.

*Table 23. Tokens available to PostUserTableCreateRunScript*

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 EMM 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.
<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****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.

*Table 24. Tokens available to PrefixOnSelectSQL*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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****Description**

The QueryThreadSleep property affects the CPU utilization of the Campaign server process (UNICA\_ACSSVR). 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**

### **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

## **SegmentTempTablePrefix**

### **Description**

Sets the prefix for Segment tables created by the CreateSeg process in this data source.

### **Default value**

UACS

## **ShareConnection**

### **Description**

The ShareConnection property is no longer used and should remain set to its default value, FALSE.

### **Default value**

FALSE

### **Valid Values**

FALSE

## **SQLOnConnect**

### **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.

*Table 25. Tokens available to SQLOnConnect*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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.

### StringEncoding

#### 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

### 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.

*Table 26. Tokens available to SuffixOnAllOtherSQL*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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.

*Table 26. Tokens available to SuffixOnAllOtherSQL (continued)*

Token	Description
<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

#### 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.

### SuffixOnInsertSQL

#### 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.

*Table 27. Tokens available to SuffixOnInsertSQL*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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.

### SuffixOnSelectSQL

#### 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.

*Table 28. Tokens available to SuffixOnSelectSQL*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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.

*Table 28. Tokens available to SuffixOnSelectSQL (continued)*

Token	Description
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

#### Default value

No default value defined.

### SuffixOnTempTableCreation

#### Description

Use the SuffixOnTempTableCreation 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.

*Table 29. Tokens available to SuffixOnTempTableCreation*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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.

### **SuffixOnSegmentTableCreation**

#### **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.

*Table 30. Tokens available to SuffixOnSegmentTableCreation*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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

### **SuffixOnSnapshotTableCreation**

#### **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.

*Table 31. Tokens available to SuffixOnSnapshotTableCreation*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM user name associated with the flowchart for which the Snapshot table was created.

*Table 31. Tokens available to SuffixOnSnapshotTableCreation (continued)*

Token	Description
<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.
<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

Valid SQL

### SuffixOnExtractTableCreation

#### 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.

*Table 32. Tokens available to SuffixOnExtractTableCreation*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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.

*Table 32. Tokens available to SuffixOnExtractTableCreation (continued)*

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

## SuffixOnUserBaseTableCreation

**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.

*Table 33. Tokens available to SuffixOnUserBaseTableCreation*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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.

### **SuffixOnUserTableCreation**

#### **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.

*Table 34. Tokens available to SuffixOnUserTableCreation*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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.
<TABLENAME>	This token is replaced with the temp table name.

#### **Default value**

No default value defined.

### **SystemTableSchema**

#### **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.

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 (for example, dbo.UA\_Folder). In this form, X is the schema and Y is the unqualified table name. This terminology for this syntax differs among the different database systems supported by Campaign.)

If multiple schemas exist in the system tables database, then set this value to the name of the schema in which the Campaign system tables were created.

#### Default value

No default value defined.

### TempTablePostExecutionSQL

#### Description

Use the TempTablePostExecutionSQL property to specify a complete SQL statement that Campaign runs immediately after the creation of a temporary table in a user data source or in the system tables database. The AllowTempTables property must be set to TRUE to enable the creation of temp tables in a data source.

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.

The TempTablePostExecutionSQL 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 the TempTablePostExecutionSQL property, be sure that you 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\;

Tokens available to TempTablePostExecutionSQL are described below.

Table 35. Tokens available to TempTablePostExecutionSQL

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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.

Table 35. Tokens available to TempTablePostExecutionSQL (continued)

Token	Description
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

#### Default value

No default value defined.

### TableListSQL

#### 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, 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 (\*), 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 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

```

## **UOSQLOnConnect**

### **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.

*Table 36. Tokens available to UOSQLOnConnect*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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.

## **UseSQLToRetrieveSchema**

### **Description**

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 for this property 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:

- For Netezza, you must set this property to TRUE to allow support for synonyms. Setting this property to TRUE tells Campaign to prepare a SQL query to retrieve the table schema. No other settings or values are needed to support synonyms in Netezza data sources.
- For SQL Server, to allow support for synonyms you must 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.
- For Oracle data sources, setting this property to TRUE tells 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`

## UserTablePostExecutionSQL

#### Description

Use the `UserTablePostExecutionSQL` property to specify a complete SQL statement that Campaign runs immediately after the creation of a user table in a user data source or in the system tables database.

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.

The UserTablePostExecutionSQL 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 the UserTablePostExecutionSQL property, be sure that you 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\;

Tokens available to UserTablePostExecutionSQL are described below.

*Table 37. Tokens available to UserTablePostExecutionSQL*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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.
<KEYCOLUMNS>	This token is replaced with the user table column name(s).
<TABLENAME>	This token is replaced with the user table name.
<USER>	This token is replaced with the Campaign user name of the user running the flowchart.

#### Default value

No default value defined.

### UseTempTablePool

#### Description

WhenUseTempTablePool 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

## SegmentTablePostExecutionSQL

### 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.

*Table 38. Tokens available to SegmentTablePostExecutionSQL*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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

A valid SQL statement

## SnapshotTablePostExecutionSQL

### 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.

Tokens available to SnapshotTablePostExecutionSQL are described below.

*Table 39. Tokens available to SnapshotTablePostExecutionSQL*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM user name associated with the flowchart for which the Snapshot table was created.

*Table 39. Tokens available to SnapshotTablePostExecutionSQL (continued)*

Token	Description
<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.
<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

### TempTablePrefix

#### Description

The TempTablePrefix parameter specifies a string that is automatically prepended to the names of all temporary tables created by Campaign. Use this parameter to help you identify and manage your 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.

Tokens available to TempTablePrefix are described below.

*Table 40. Tokens available to TempTablePrefix*

Token	Description
<AMUSER>	This token is replaced with the IBM EMM 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.

*Table 40. Tokens available to TempTablePrefix (continued)*

Token	Description
<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.

**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.

#### Default value

UAC

### **TempTablePreTruncateExecutionSQL**

#### Description

**Note:** This property is supported only for Teradata data sources. For all other supported databases, this property should not be set.

Use the TempTablePreTruncateExecutionSQL 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**

#### Description

**Note:** This property is supported only for Teradata data sources. For all other supported databases, this property should not be set.

Use the TempTablePreTruncateRunScript 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

#### Description

The TeradataDeleteBeforeDrop parameter applies only to Teradata data sources. It specifies whether records are deleted before a table is dropped.

Set this value to TRUE to delete all records from a table before dropping the table.

**Note:** If Campaign is unable to delete the records for any reason, it will not drop the table.

Set this value to FALSE to drop a table without first deleting all records.

#### Default value

TRUE

### TruncateSQL

#### Description

The TruncateSQL 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 set to TRUE. When DeleteAsTruncate is set to TRUE, any custom SQL in this property is used to truncate a table. When this property is not set, Campaign uses the TRUNCATE TABLE <TABLENAME> syntax.

This parameter is undefined by default.

Tokens available to TruncateSQL are described below.

*Table 41. Tokens available to TruncateSQL*

Token	Description
<TABLENAME>	This token is replaced with the database table name that Campaign is truncating.

#### Default value

No default value defined.

### Type

#### Description

The partitions > partition[n] > dataSources > [data\_source\_name] > type property specifies the database type of this data source.

#### Default value

The default value depends on the database template used to create the data source configuration.

#### Valid Values

Valid values for system tables are:

- SQLServer
- DB2
- DB2ODBC
- ORACLE
- ORACLE8
- ORACLE9

Valid values for customer tables also include:

- TERADATA
- NETEZZA

### UseExceptForMerge

#### Description

When 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 set to TRUE and we 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)
```

#### Default value

FALSE

#### Valid Values

TRUE | FALSE

### UseMergeForTrack

#### Description

Implements SQL MERGE syntax to improve the performance of the Track process. The UseMergeForTrack 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

#### 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

#### Description

When 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 (value set to YES), 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

NO

#### Valid Values

YES | NO

### UseSQLToProfile

#### Description

The UseSQLToProfile property allows you to configure 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 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 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

### **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 | 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**

##### **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.

Note that in this version of Campaign, the NumberOfferCodesToUse parameter no longer exists. This value now comes from the offer template (every offer template can have a different number of offer codes).

#### **Default value**

-

#### **allowVariableLengthCodes**

##### **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**

### **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**

### **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**

### **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 as follows:

- 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**

### **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**

#### Description

The campCodeGenProgFile property specifies the name of the campaign code generator.

The properties that control the format of the code generated 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, and 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)

### **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 EMM 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**

**collaborateInactivityTimeout**

**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

## **Campaign | partitions | partition[n] | server | spss**

Properties in this category affect IBM SPSS Modeler Advantage Marketing 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 Marketing Edition in this directory.
- IBM SPSS Modeler Advantage Marketing 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 Marketing 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 Marketing 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 set on folders created by Campaign, and the UNIX group and permissions set on files contained in the **profile** directory.

### **userFileGroup (UNIX only)**

#### **Description**

The **userFileGroup** 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

### **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 EMM 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 to TRUE.

**Default value**

FALSE

**Valid Values**

TRUE | FALSE

**Campaign | partitions | partition[n] | server | flowchartSave**

The properties in this category specify the default settings for a new Campaign flowchart's auto-save and checkpoint properties.

**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 Campaign server optimization for partitions.

**Note:** This category of parameters is not related to Contact Optimization.

## **maxVirtualMemory**

### **Description**

The `maxVirtualMemory` property specifies a default setting for a new Campaign flowchart's Campaign Virtual Memory Usage Limit property, configurable for each flowchart through the client-side Advanced Settings window. The units are in megabytes.

### **Default value**

128

## **useInDbOptimization**

### **Description**

The `useInDbOptimization` property specifies whether Campaign tries to perform as many operations as possible in the database instead of in the Campaign server.

If the value is FALSE, Campaign maintains lists of IDs in the Campaign server at all times.

If the value is TRUE, Campaign avoids pulling the ID lists if possible.

### **Default value**

FALSE

### **Valid Values**

TRUE | FALSE

## **maxReuseThreads**

### **Description**

The `maxReuseThreads` 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 as this property is set to 0.

Use the cache to reduce the overhead of thread allocation or if your operating system exhibits 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**

#### **Description**

The `threadStackSize` 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**

#### **Description**

The `tempTableDataSourcesForSegments` 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**

#### **Description**

To improve performance for strategic segments, set this option to TRUE. When this option is set to TRUE, strategic segments do not create binary files on the Campaign server; instead, strategic segments create Segment temp tables in the data source. When the value is set to TRUE, at least one valid Temp Table data source must be specified in the CreateSeg process configuration.

#### **Default value**

FALSE

#### **Valid Values**

TRUE | FALSE

### **forceViewForPreOptDates**

#### **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

### Campaign | partitions | partition[n] | server | logging

Properties in this category specify, for the Campaign server, whether standard and Windows event logging are enabled, logging levels and categories, and other logging behavior.

#### enableWindowsEventLogging

##### Description

The enableWindowsEventLogging property enables or disables 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. If disabled, the windowsEventLoggingLevel and windowsEventLoggingCategory settings are ignored.

#### Default value

FALSE

#### Valid Values

TRUE | FALSE

### logFileBufferSize

##### Description

The logFileBufferSize property is used when the value of the keepFlowchartLogOpen property is yes. It sets an upper bound on the number of log messages after which the messages will be written to file.

If the value is 1, every log message is written immediately to file, effectively disabling buffering but causing somewhat worse performance.

This property is ignored if the value of keepFlowchartLogOpen is set to no.

#### Default value

5

### keepFlowchartLogOpen

##### Description

The keepFlowchartLogOpen property specifies whether Campaign opens and closes the flowchart log file each time a line is written to the log file.

If the value is FALSE, Campaign opens and closes the flowchart log file.

If the value is TRUE, Campaign opens the flowchart log file only once, and closes the flowchart log file only when the flowchart's server process exits. A value of yes may improve performance of real-time flowcharts. A side effect of using the yes setting is that recently-logged messages may not be immediately visible in the log file, as 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.

#### **Default value**

FALSE

#### **Valid Values**

TRUE | FALSE

### **logProcessId**

#### **Description**

The `logProcessId` property controls whether the process ID (pid) of the Campaign Server process is logged 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**

#### **Description**

The `logMaxBackupIndex` 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**

#### **Description**

The `loggingCategories` property specifies the category of messages written to the Campaign server log file. This works in conjunction with `loggingLevels`, which determines which messages are logged based on severity (for all selected categories). You can specify multiple categories in a comma-separated list. The special category `all` provides a shorthand for specifying all logging categories.

**Default value**

ALL

**Valid Values**

Supported categories are:

- 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

**loggingLevels****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
- MEDIUM
- HIGH
- ALL

LOW represents the least detail (the most severe errors only), and ALL includes trace messages and is intended primarily for diagnostic purposes. You can adjust these settings from within a flowchart through the Tools > Logging Options menu.

**Note:** You may want to set the `loggingLevels` property to ALL during configuration and testing, to maximize the logging output from Campaign for diagnostic purposes. This setting generates a large amount of data and therefore may not be advisable for production operation.

## **windowsEventLoggingCategories**

### **Description**

The windowsEventLoggingCategories property specifies the category of messages written to the Campaign server windows event log. This 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 special 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**

### **Description**

The logFileMaxSize 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**

### **Description**

The windowsEventLoggingLevel property controls the amount of detail written to the Campaign server windows event log based on severity.

### **Default value**

MEDIUM

**Valid Values**

- LOW
- MEDIUM
- HIGH
- ALL

LOW represents the least detail (the most severe errors only), and ALL includes trace messages and is intended primarily for diagnostic purposes.

**enableLogging****Description**

The enableLogging property specifies whether 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****Description**

The AllowCustomLogPath property allows user to change the log path for each campaign flowchart that generates flowchart-specific logging information when it is run.

A setting of False prevents the user from changing the path to which the flowchart log file is written.

A setting of True enables the user to change the path through the user interface or when using unica\_svradm to run the flowchart.

**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**

The saveRunResults property specifies what files are saved when you save an Campaign flowchart.

If the value is TRUE, the “underscore” files are saved and, if the value of useInDbOptimization is yes, database temp tables persist.

If the value is FALSE, only the .ses file is saved and you cannot view intermediate results if you reload the flowchart.

### **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 an 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 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**

#### **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**

#### **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**

#### **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**

#### **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**

#### 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**

#### 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**

#### 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**

#### 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**

**Description**

Enables IBM 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 **IBM Digital Analytics Segments** as input. To configure the IBM Digital Analytics integration for each partition, choose **Settings > Configuration > Campaign | partitions | partition[n] | Coremetrics**.

**Default value**

No

**Valid Values**

Yes | No

## **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\_ACDDIR**.

**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\_ACDDIR**.

**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 | partitions | partition[n] | Coremetrics

Properties in this category specify integration settings for IBM Digital Analytics and Campaign for the selected Campaign 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 IBM Digital Analytics integration service that provides the integration point between IBM Digital Analytics and Campaign.

#### 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 IBM 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 IBM 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 IBM 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 IBM 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

This property specifies which IBM EMM account is allowed to access the IBM 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

This 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 IBM 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

## Campaign I 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**

The `Campaign | unicaACListener` properties specify logging levels, some access privileges, language encodings, number of operating system threads, and the protocol, host, and port of the Campaign listener. These properties must be set only once per instance of Campaign; they do not need to be set for every partition.

### **enableWindowsImpersonation**

##### **Description**

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**

##### **Description**

The Campaign > unicaACListener > enableWindowsEventLogging property controls logging to the Windows event log. Set this property to TRUE to log to the Windows event log.

**Default value**

FALSE

**Valid Values**

TRUE | FALSE

**serverHost**

**Description**

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 EMM 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**

**Description**

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 starts a new log file named File.

**Default value**

1 (creates one backup file)

**logStringEncoding**

**Description**

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**

### **Description**

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**

### **Description**

The `Campaign > unicaACListener > loggingLevels` property controls the amount of detail written to the log file.

### **Default value**

`MEDIUM`

### **Valid Values**

- `LOW`
- `MEDIUM`
- `HIGH`

## **maxReuseThreads**

### **Description**

The `Campaign > unicaACListener > maxReuseThreads` 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.

### **Default value**

`0` (zero), which disables the cache

## **logMaxFileSize**

### **Description**

The `logMaxFileSize` property specifies the maximum size, in bytes, that the log file can reach before rolling into the backup file.

### **Default value**

`10485760` (10 MB)

## **windowsEventLoggingLevels**

### **Description**

The windowsEventLoggingLevel's property controls the amount of detail written to the Windows event log file based on severity.

### **Default value**

MEDIUM

### **Valid Values**

- LOW
- MEDIUM
- HIGH
- ALL

The ALL level includes trace messages intended for diagnostic purposes.

## **serverPort**

### **Description**

The serverPort property specifies the port where the Campaign listener is installed.

### **Default value**

4664

## **useSSL**

### **Description**

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**

### **Description**

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**

#### **Description**

See the description for the `serverPort2` property, in this category.

#### **Default value**

FALSE

#### **Valid Values**

TRUE, FALSE

### **keepalive**

#### **Description**

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

### **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 the 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.

### **Campaign | logging**

The property in this category specifies the location of the Campaign log properties file.

#### **log4jConfig**

##### **Description**

The `log4jConfig` property specifies the location of the Campaign log properties file, `campaign_log4j.properties`. Specify the path relative to the `Campaignhome` directory, including the file name. Use forward slashes (/) for UNIX and backslashes (\) for Windows.

##### **Default value**

`./conf/campaign_log4j.properties`

---

## **eMessage configuration properties**

This section describes the eMessage configuration properties found on the Configuration page.

### **eMessage | serverComponentsAndLocations | hostedServices**

Properties on this page specify the URLs for connecting to IBM EMM 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 EMM Hosted Services through the data center that IBM has established 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 EMM Hosted Services, except uploading recipient lists and related metadata.

**Default value**

em.unicaondemand.com

If you are connecting to IBM's U.K. data center, change this value to em-eu.unicaondemand.com.

**dataHostName**

**Description**

The address that eMessage uses for uploading metadata related to recipient lists to IBM EMM Hosted Services.

**Default value**

em.unicaondemand.com

If you are connecting to IBM's U.K. 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 EMM Hosted Services.

**Default value**

ftp-em.unicaondemand.com

If you are connecting to IBM's U.K. data center, change this value to ftp-em-eu.unicaondemand.com.

**eMessage | partitions | partition[n] | hostedAccountInfo**

Properties in this category allow you to define user credentials required to access the database used to store account information required to access IBM EMM Hosted Services. Values 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 required to access IBM EMM 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 EMM Hosted Services.

**Default value**

UNICA\_HOSTED\_SERVICES

**Valid Values**

A data source 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 EMM 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 EMM 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 PreparedStatement 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 performs an action 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 EMM Hosted Services. For example, you can create a script to send an email alert to the list designer when the list upload has completed successfully.

If you define a value for this property, eMessage will pass status information about the Recipient List Uploader to the specified location. eMessage takes 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 EMM Hosted Services. This property allows you to instruct the RCT to temporarily stop retrieving data from IBM EMM 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 EMM 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 EMM 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 EMM 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.

**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.
<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 SQLSRV 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.

Token	Description
<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**

**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 | [ExternalCalloutName]:**

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.

**Interact | flowchart | ExternalCallouts | [ExternalCalloutName] | Parameter Data | [parameterName]:**

This section defines the parameter settings for a custom external callout you have written with the external callout API.

**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 I 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 IBM EMM > 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 IBM EMM > 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 IBM EMM > 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.

### **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, even if `enableOfferSuppressionLookup` is set to true.

Interact looks for this table in the customer tables accessed by runtime servers, defined by the `prodUserDataSource` properties.

**Default value**

`UACI_BlackList`

**profileTable****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 name of the table containing the profile data for this audience level.

Interact looks for this table in the customer tables accessed by 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_usrProd`. If you enter a fully-qualified name, for example, `mySchema.UACI_usrProd`, Interact does not prepend the schema name.

**Default value**

No default value defined.

**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.

**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.

**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.

**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.

**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.

**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

*Interact | profile | Audience Levels | [AudienceLevelName] | SQL Template:*

These configuration properties let you define one or more SQL query templates used by the offersBySQL feature of Interact.

**name****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****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

## **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

## **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 setting 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.

## **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

## **aggregateStatsIntervalInMinutes**

### **Description**

The number of minutes the Interact learning module waits between aggregating data in 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**

15

**Valid Values**

An integer greater than zero.

**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 has been collected for recommendations.

**Default value**

False

**Valid Values**

True | False

**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 | 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 responsHist 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 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 Opportunity Detection is enabled, and they define the timeout intervals for connections with Interact Opportunity Detection.

**enableAdvancedPatterns****Description**

If true, enables integration with Interact Opportunity Detection. If false, integration is not enabled. If integration was previously enabled, Interact uses the most recent pattern states received from Interact Opportunity Detection.

**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 Opportunity Detection. 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 Opportunity Detection, and a request is sent to the Interact Opportunity Detection 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 Opportunity Detection.

**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 Opportunity Detection.

**enable****Description**

Determines whether the system to reconnects automatically if connection problems occur between the Interact real time environment and Interact Opportunity Detection. 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 evaluates repeated connection problems occurring between the Interact real time environment and Interact Opportunity Detection.

#### 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 Opportunity Detection.

#### 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 Opportunity Detection. 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 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 | 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 xsesssion 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****use.StoredProcedure****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**

**use.StoredProcedure**

**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

<b>Type</b>	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.
<b>Default value</b>	2
<b>Valid Values</b>	An integer defined in the UACI_TrackingType table.
<b>Interact   services   crossSessionResponse   OverridePerAudience   [AudienceLevel]   TrackingCodes   byAlternateCode:</b>	The properties in this section define how cross-session response tracking matches a user-defined alternate code to contact and response history.
<b>Name</b>	
<b>Description</b>	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.
<b>Default value</b>	
<b>OverrideSQL</b>	
<b>Description</b>	The SQL command or stored procedure to match the alternate code to the contact and response history by offer code or treatment code.
<b>Default value</b>	
<b>useStoredProcedure</b>	
<b>Description</b>	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.
<b>Default value</b>	false
<b>Valid Values</b>	true   false
<b>Type</b>	
<b>Description</b>	The associated TrackingCodeType defined in the UACI_TrackingType table in the runtime environment tables.
<b>Default value</b>	

## **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 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 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 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 write cached data to the data store.

**Default value**

5

**Interact | services | configurationMonitor:**

The configuration properties in this category allow you to enable or disable integration with Interact Opportunity Detection 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 file 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 file is automatically created in the operating system's default temporary directory.

#### Default value

None

#### Valid Values

A file 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/EMM/Interact/lib/interact_wxs_adapter.jar` or

`file:///C:/IBM/EMM/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.

**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.

**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.

**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.

#### 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 **jndiSettings** category and click the *(jndiSetting)* property.

##### (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.
- **jndiName** 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 EMM user that has been configured with the credentials for connecting to the database in this JDBC connection.
- **asmDataSource** the name of IBM EMM 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 to the database in this JDBC connection.

#### Default value

None.

## Interact design environment configuration properties

This section describes all the configuration properties for Interact design environment.

### Campaign | partitions | partition[n] | reports

These configuration properties define 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 using 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 using 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 using 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 using 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 using 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 using 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 using 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 using 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 have installed 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 have installed 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, four arguments are passed to the completion notification script: start time, end time, total number of CH records processed, and total number of RH records processed. The start time and end time are numeric values representing number of milliseconds elapsed since 1970.

**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**

**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\_Dt1ContactHist 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 > offersisting > 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.

### **enableLearning**

#### **Description**

If set to Yes, the Interact design time expects learning to be enabled. If you set `enableLearning` to yes, you must configure `Interact > offersisting > optimizationType` to `BuiltInLearning` or `ExternalLearning`.

If set to No, the Interact design time expects learning to be disabled. If you set `enableLearning` to no, you must configure `Interact > offersisting > optimizationType` to `NoLearning`.

#### **Default value**

No

### **maxAttributeNames**

#### **Description**

The maximum number of learning attributes the Interact learning utility monitors.

This property is applicable if the `Interact > offersisting > 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 > offersisting > 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 > offersisting > 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. When learning is enabled, enabling percentRandomSelection overrides the offerTieBreakMethod configuration property.

**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 > offersisting > 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] | DefaultOffers**

These configuration properties define the default cell code for the default offers table. You need to configure these properties only if you are defining global offer assignments.

### **DefaultCellCode**

#### **Description**

The default cell code Interact uses if you do not define a cell code in the default offers table.

#### **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 | whiteList | [AudienceLevel] | offersBySQL**

These configuration properties define the default cell code for the offersBySQL table. You need to configure these properties only if you are use SQL queries to get a desired set of candidate offers.

### **DefaultCellCode**

#### **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.

#### **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 | whiteList | [AudienceLevel] | ScoreOverride**

These configuration properties define the default cell code for the score override table. You need to configure these properties only if you are defining individual offer assignments.

### **DefaultCellCode**

#### **Description**

The default cell code Interact uses if you do not define a cell code in the score override table.

#### 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] | 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

##### 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

##### 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

##### 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**

### **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**

### **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**

### **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**

### **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**

**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**

**Description**

Enables IBM 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 **IBM Digital Analytics Segments** as input. To configure the IBM Digital Analytics integration for each partition, choose **Settings > Configuration > Campaign | partitions | partition[n] | Coremetrics**.

**Default value**

No

**Valid Values**

Yes | No

**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 effects 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 effects 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 effects 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 | 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

Since 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 which 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

## **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

## **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

## **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

**EnableBufferingHistoryTransactions****Description**

If true, Contact Optimization writes contact history transactions to a file to read during an 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 MinImprovementPercent 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 which triggered the generation loop error. Contact Optimization then continues processing the next customer set in the Contact Optimization session. 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 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. By default, all rows are logged if you set this value to yes.

If you do not want processed rows of the proposed contacts table 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 an Optimize 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 Optimize.

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.

## **Campaign | partitions | partition[n] | Optimize | logging**

This property defines logging settings for Contact Optimization.

### **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

### 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

### 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

### 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

## **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

## **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

## **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)

---

## **Distributed Marketing configuration properties**

This section describes the Distributed Marketing configuration properties on the configuration page.

Additional configuration properties exist in XML files 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. You should not change this value.

#### **Default Value**

Collaborate

### **httpPort**

#### **Description**

The port number used by the application server for connections to the Distributed Marketing application.

#### **Default Value**

7001

### **httpsPort**

#### **Description**

The port number 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.

**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`

**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 executions in order 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 Flowcharts Run 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 using the wizard. If there are any required fields that are 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 username to login 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 login 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 login 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 with respect to 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 FTP server 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 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**

Username to login to SFTP server.

**Default value**

No default value defined.

**collaborateAttachmentsSFTP\_password**

**Description**

Optional. The SFTP password to login 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 login on SFTP server.

**Default value**

No default value defined.

**collaborateAttachmentsSFTP\_keypassphrase**

**Description**

SFTP passphrase to login 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 with respect to 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

**mergeFullPath**

**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 will validate 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 execute 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 execute 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-folder-id> tag in a project template file.

#### **Default value**

2

## **Collaborate | UDM Configuration Settings | Datasource**

These configuration properties are for Datasource settings.

**jndiName****Description**

Datasource name for Distributed Marketing database.

**Default value**

collaborateds

**asmJndiName****Description**

Datasource 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 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 issuing 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 9.0.0 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 issuing a timeout error.

### **Default value**

600000

## **flowchartServiceScheduleServices10MaxRetries**

### **Description**

The number of times Distributed Marketing attempts to connect with the Marketing Platform scheduler before issuing an error.

### **Default value**

3

## **flowchartServiceScheduleServices10RetryPollPeriod**

### **Description**

The number of seconds Distributed Marketing waits before attempting 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 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 which are sent to Campaign by the isAlive monitor before throwing a flowchart execution error.

**Default value**

10

**flowchartServicesAliveMonitorPollPeriod**

**Description**

The time, in seconds, to wait between queries 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 charge comments when saving 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**

## **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 defined in the procedure-plugins.xml.

#### **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 will be returned on the search screen.

**Default value**

1000

### **listManagerListPageSize**

**Description**

The number of rows displayed on a page in List Manager.

**Default value**

20

### **listManagerListsMaxrow**

**Description**

The maximum number of rows displayed in a list.

**Default value**

10000

## **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 week days in terms of counts, such as Sunday = 1, Monday = 2, etc. 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

### **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

**notifyDefaultSenderIdEmailAddress****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]

**notifySenderIdAddressOverride****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 will be 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 will be 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 returned by 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. Read-only lookup table names. The lookup table may not be updated in the Form Editor, and is allowed as a wildcard at the end of lookup table names.

**Default value****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`

## **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\DistributionMarketing\templates.

#### **Default value**

`templates`

### **projectTemplatesFile**

#### **Description**

The specified file describes the various sort of projects: List, On-demand, and Corporate Campaign.

#### **Default value**

`project_templates.xml`

### **templateAutoGenerateNameEnabled**

#### **Description**

Indicates if template name for new template must be generated or not.

#### **Default value**

`true`

#### **Valid values**

`TRUE | FALSE`

### **defaultListTableDSName**

#### **Description**

Used to assign datasource name for templates while importing template if datasource name is not defined.

#### **Default value**

`ACC_DEMO`

## **templateAdminGroup\_Name**

### **Description**

Specifies multiple groups. Users belonging to these groups will 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 (valid values are 0-12, representing midnight to noon). This setting is used as the denominator when calculating a percentage of task completion in workflows.

### **Default value**

9

### **numberOfHoursPerDay**

### **Description**

Indicates the number of hours per day (valid values are 1-24). The default indicates a standard, 8-hour work day. This setting is used as the denominator when calculating a percentage of task completion in workflows.

### **Default value**

8

### **automaticallyRestartFailedRecurrentTasks**

### **Description**

Decides whether to automatically restart the failed recurrent tasks. If the value of parameter is set to false, users will have to manually update the

failed task status to Pending either from workflow or from post-task update pop. The schedule will pick up only those tasks for execution which are in pending state.

If the value is set to true, no manual intervention will be 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 Marketing Edition configuration properties

Properties in this category specify values that are used to configure IBM EMM for single sign-on with IBM SPSS Modeler Advantage Marketing Edition.

See the *IBM Campaign and IBM SPSS Modeler Advantage Marketing Edition Integration Guide* for complete instructions on setting up single sign-on with IBM SPSS Modeler Advantage Marketing Edition.

### SPSS I integration

Properties in this category are used to configure Marketing Platform for single sign-on with IBM SPSS Modeler Advantage Marketing Edition.

#### Platform user for IBM SPSS account

**Description**

Enter the login name for the IBM SPSS Modeler Advantage Marketing Edition account that you created or identified for single sign-on with IBM SPSS Modeler Advantage Marketing Edition.

**Default value**

asm\_admin

**Availability**

This property is used only to configure Marketing Platform for single sign-on with IBM SPSS Modeler Advantage Marketing 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 Marketing 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 Marketing 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 Marketing 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 Marketing Edition.

### Enable IBM SPSS

#### Description

Set this property to TRUE to enable single sign-on with IBM SPSS Modeler Advantage Marketing 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 Marketing Edition.

## SPSS | navigation

Properties in this category affect IBM SPSS Modeler Advantage Marketing 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 Marketing 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 Detection configuration properties

This section describes the Opportunity Detection configuration properties on the Configuration page.

### Opportunity Detection

Properties in this category specify the default locale for Opportunity Detection and Interact Opportunity Detection.

#### **currencyLocale**

**Description**

The locale that governs the way currency is displayed and stored in Opportunity Detection.

**Default value**

`en_US`

## **supportedLocales**

### **Description**

A list of the locales supported for data storage in this version of Opportunity Detection. Changes to this value have no effect and are not recommended.

### **Default value**

de,en,fr,ja,es,ko,pt,it,zh

## **defaultLocale**

### **Description**

The assumed locale for all Opportunity Detection users. Changes to this value are not recommended.

### **Default value**

[default-locale]

## **Opportunity Detection | Navigation**

Properties in this category specify values that are used internally to navigate among IBM EMM products.

## **welcomePageURI**

### **Description**

The Uniform Resource Identifier of the IBM Opportunity Detection index page. This value is used internally by IBM EMM applications. Changes to this value are not recommended.

### **Default value**

/index.jsp

## **seedName**

### **Description**

Used internally by IBM EMM applications. Changes to this value are not recommended.

### **Default value**

Detect

## **type**

### **Description**

Used internally by IBM EMM 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 Detection application.

**Default value**

7001

**httpsPort**

**Description**

The port number that is used by the application server for secure connections to the Opportunity Detection application.

**Default value**

7001

**serverURL**

**Description**

The URL of the Opportunity Detection installation. Accepts either the HTTP or HTTPS protocol. You must use the domain and the machine name (rather than the IP address) when IBM EMM products are installed on more than one machine.

**Important:** If IBM EMM 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 EMM.

**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 Detection

## Opportunity Detection | System | Interact Design Service

The property in this category specifies the URL of the Interact web service, used when Interact is integrated with Interact Opportunity Detection.

## **ServerURL**

### **Description**

The URL for the Interact web service. For example, `http://IP_address:port/axis2/services/InteractDesignService`.

### **Default value**

`http://[InteractServiceHost]:[InteractServicePort]/axis2/services/InteractDesignService`

## **Opportunity Detection | System | Interact Connector**

The property in this category specifies the URL for the web service used when Interact is integrated with Interact Opportunity Detection.

## **ServerURL**

### **Description**

The URL for the Interact web service. For example, `http://IP_address:port/servlets/StreamServlet`.

### **Default value**

`http://[InteractConnectorHost]:[InteractConnectorPort]/servlets/StreamServlet`

## **Opportunity Detection | logging**

The property in this category specifies the location of the Opportunity Detection log file.

## **log4jConfig**

### **Description**

The location of the configuration file that Opportunity Detection uses for logging. If you change this path, you must restart the web application server to apply the change.

### **Default value**

`[absolute-path]/conf/detect_log4j.properties`



---

## Appendix B. Re-branding the frameset

You can customize the appearance of the user interface where most IBM EMM 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.

---

## To prepare your corporate theme

Follow these guidelines to create your corporate theme for the IBM EMM frameset.

### Procedure

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.

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.

---

## Applying your corporate theme

Follow this procedure to apply your corporate theme to the IBM EMM user interface.

### Procedure

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 “To prepare your corporate theme” on page 511 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 “To prepare your corporate theme” on page 511, 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 EMM.  
Your new theme should be applied.

---

## 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.

### 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](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.



---

## 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 about 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 websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

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  
170 Tracer Lane  
Waltham, MA 02451  
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](http://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."





**IBM**<sup>®</sup>

Printed in USA