

Unica Platform V12.1.7 Administrator's Guide



Contents

- Chapter 1. Administrator Guide..... 8**
- Unica Platform features..... 8
 - About Unica Platform security features..... 8
 - Configuration management..... 10
 - Localization in Unica..... 10
 - The common user interface..... 10
 - Logging in to Unica..... 11
 - Unica Platform documentation and help..... 12
- New Navigation UI..... 13
- What's New..... 16
- Licensing - overview..... 16
 - Licensing portal..... 16
- Unica user account management..... 19
 - Types of user accounts: internal and external..... 19
 - Properties of internal user accounts..... 20
 - Adding internal user accounts..... 20
 - Deleting internal user accounts..... 21
 - Changing internal user password expiration dates..... 21
 - Resetting internal user passwords..... 21
 - Changing internal user account properties..... 22
 - Changing internal user system status..... 22
 - Changing internal user type..... 22
 - Adding internal user data sources..... 23
 - Changing internal user data sources..... 23
 - Deleting internal user data sources..... 23
 - The user management pages..... 24
 - Locale preference..... 26
 - Synchronization of external users..... 27
- Security management..... 27
 - Permissions and tasks of the security administrator in Unica Platform..... 28

Special characters in role and policy names.....	28
Roles and permissions in Unica Platform and Unica Campaign.....	28
Overview of managing user application access in Unica Platform	29
Types of groups: internal and external.....	29
Partitions and security management.....	30
Pre-configured users and roles.....	31
Cross-partition administration privileges.....	32
Adding an internal group.....	33
Adding a subgroup.....	33
Deleting a group or subgroup.....	33
Changing a group or subgroup description.....	34
Assigning a group to a partition.....	34
Adding a user to a group or subgroup.....	34
Removing a user from a group or subgroup.....	35
The user group management pages.....	35
Creating a role.....	37
Modifying role permissions.....	37
Removing a role from the system.....	38
Assigning a role to or removing a role from a group.....	38
Assigning a role to or removing a role from a user.....	38
Definitions of permission states.....	39
Permissions for products that use only basic roles.....	39
Permissions for Unica Platform	41
Permissions for Opportunity Detect.....	42
Configuration management.....	43
Property categories.....	43
Property descriptions.....	44
The refresh function.....	45
The default user locale preference.....	45
Navigating to a category.....	45
Editing property values.....	46
Creating a category from a template.....	46

Deleting a category.....	46
Dashboard management.....	47
Dashboard planning.....	47
Dashboard audiences.....	47
User permissions required to view dashboards.....	48
Pre-defined portlets.....	48
Pre-assembled dashboards.....	53
IBM® Cognos® report performance considerations.....	55
Dashboard setup.....	57
Quick link portlets.....	62
Custom portlets.....	63
Dashboard membership administration.....	68
The Unica Scheduler.....	69
Scheduler triggers that are sent on success or failure of runs.....	70
Schedules that depend on completion of multiple runs.....	71
Schedule triggers that are sent from an external script.....	72
Scheduler recurrence patterns.....	73
Time zone support.....	73
Scheduler throttling.....	74
Whitelist prerequisite for external tasks (with FixPack 10.0.0.1 only).....	75
Best practices for setting up schedules.....	76
To create a schedule wizard.....	77
Run exclusions.....	81
What to consider when you use the scheduler with Unica Campaign.....	86
Schedule notifications.....	90
Schedule management.....	91
SAML 2.0 based federated authentication.....	98
How to implement federated authentication.....	100
Related concepts.....	109
SAML 2.0 single sign-on.....	110
Behavior when SAML 2.0 single sign-on is implemented.....	111
Configuration process roadmap: SAML 2.0 single sign-on.....	112

Setting up the metadata file.....	112
Setting SAML 2.0 configuration properties.....	113
Setting up a data source for SAML single sign-on.....	113
Sample SAML 2.0 IdP assertion.....	113
Sample IdP metadata.....	117
Single sign-on between Unica and IBM Digital Analytics	119
Setting up single sign-on between Unica and Digital Analytics using automatic user account creation.....	120
Setting up single sign-on between Unica and Digital Analytics using manual user account creation.....	122
Configuring WebLogic for single sign-on between Digital Analytics and Unica.....	123
Configuring WebSphere® for single sign-on between Digital Analytics and Unica.....	123
Digital Analytics integration with Websense using a custom proxy.....	124
Integration between Unica and Windows™ Active Directory.....	126
Active Directory integration features.....	127
Active Directory integration prerequisites.....	129
Configuration process roadmap: Active Directory integration.....	129
Integration between Unica and LDAP servers.....	138
LDAP integration features.....	138
LDAP integration prerequisites.....	140
Configuration process roadmap: LDAP integration.....	140
Integration with web access control platforms.....	148
About context roots.....	149
SiteMinder integration prerequisites.....	150
IBM Security Access Manager integration prerequisites.....	153
Configuration process roadmap: integrating Unica with a web access control system.....	160
Configuring integration with an SSL type of WebSEAL junction.....	162
Alert and notification management.....	163
Alert and notification subscriptions.....	163
Configuring email notifications in Unica.....	164
Implementation of one-way SSL.....	164
Overview of SSL certificates.....	165
Client and server roles in Unica.....	166
SSL in Unica	167

Configuration process roadmap: implementing SSL in Unica.....	168
Certificates for SSL.....	168
Configure your web application servers for SSL.....	174
Configure HCL Unica for SSL.....	175
Verifying your SSL configuration.....	184
Useful links for SSL.....	185
Quality of protection (QoP) settings for WebLogic.....	185
Quality of protection (QoP) settings for WebSphere.....	185
Security framework for Unica APIs.....	186
Data filter creation and management.....	189
Overview of data filter creation.....	189
Configuration process roadmap: creating data filters.....	191
Data filter XML reference.....	195
Example: Manually specifying data filters.....	202
Example: Automatically generating a set of data filters.....	208
About assigning users and groups in the XML.....	215
About assigning user and groups though the user interface.....	222
Adding data filters after the initial set has been created.....	224
Audit event tracking in Unica.....	224
Limitations on audit event tracking.....	225
Legacy audit events.....	225
Retroactive changes.....	226
Permissions for viewing the Audit Events report in a multi-partition environment.....	226
Enabling and disabling event auditing.....	226
Configuring which audit events appear in the report.....	226
Modifying the audit report content and display.....	227
Fields in the Report Parameters window.....	228
Fields and buttons in the Audit Events report.....	228
Archived audit events.....	229
Configuring audit backup notifications.....	230
Exporting the Audit Events report.....	230
Optimizing the export of the Audit Events report for large event volumes.....	231

The Unica Platform system log.....	231
System log configuration.....	232
Enabling single-user logging.....	234
Unica Platform utilities.....	238
To set up Platform utilities on additional machines.....	240
Utilities.....	241
Unica Platform SQL scripts.....	258
ManagerSchema_DeleteAll.sql.....	258
ManagerSchema_PurgeDataFiltering.sql.....	258
ManagerSchema_DropAll.sql.....	259
SQL scripts for creating system tables.....	259
Unica configuration properties.....	261
Unica Platform configuration properties.....	261
Digital Analytics configuration properties.....	333
Report configuration properties.....	334
Unica Plan configuration properties.....	335
Unica Campaign configuration properties.....	335
Unica Deliver configuration properties.....	335
Unica Interact configuration properties.....	335
Unica Journey configuration properties.....	335
Unica Content Integration configuration properties.....	335
Unica Collaborate configuration properties.....	335
IBM SPSS Modeler Advantage Enterprise Marketing Management Edition configuration properties.....	335
Opportunity Detect and Unica Interact Advanced Patterns configuration properties.....	338
Customization of stylesheets and images in the Unica user interface.....	344
Preparing your corporate theme.....	345
Applying your corporate theme.....	345
Index.....	

Chapter 1. Administrator Guide

Unica Platform features

Unica Platform provides security, configuration, notification, and dashboard features for Unica products.

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

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

About Unica Platform security features

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

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

Security technologies used in Unica Platform

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

Permission management through roles

Unica Platform defines the user's basic access to the functions within most Unica applications. In addition, for Unica Campaign and Unica 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 Unica Campaign partitions

Unica Platform provides support for partitions in the Unica Campaign family of products. Partitions provide a way to secure the data associated with different groups of users. When you configure Unica Campaign or a related Unica

application to operate with multiple partitions, each partition appears to application users as a separate instance of the application, with no indication that other partitions exist on the same system.

About groups

A subgroup inherits the roles assigned to its parents. An administrator can define an unlimited number of groups, and any user can be a member of multiple groups. This makes it easy to create different combinations of roles. For example, a user could be an Unica Deliver administrator and a Unica 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 application that requires access to a data source.

Integration with external user and group management systems

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

SAML 2.0 support

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

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

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

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

- An identity provider server WAR file.
 - A client JAR file that you can use with Java™ applications to generate and parse SAML 2.0 assertions. The Java™ products that you integrate with Unica use the assertions to communicate with the identity provider server.
- SAML 2.0 single sign-on

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

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

Users who are logged in to any application that uses the IdP server for authentication can access HCL Unica without logging in again.

Data filters

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

Configuration management

The Configuration page provides access to the central configuration properties for Unica applications.

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

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

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

Localization in Unica

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

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

Unica 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 Unica Platform also depends on the configuration of the following:

- Unica Platform system table database
- The client machines and browsers used to access Unica.

The common user interface

Unica Platform provides a common access point and user interface for Unica applications.

The common interface provides the following features.

- When multiple Unica 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 Unica 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 Unica Campaign, a search would take place across campaigns. If you wanted to search for a Unica Plan project, you would perform the search while viewing a list of Unica Plan projects.

Logging in to Unica

Use this procedure to log in to Unica.

You need the following.

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

The URL is:

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

where

host is the machine where Unica Platform is installed.

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

port is the port number where the Unica Platform application server is listening.



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

Access the Unica URL using your browser.

- If Unica 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 Unica 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, Unica displays the default dashboard page.

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



Note: Starting with version 12.1.4 Unica Campaign has migrated to OpenSSL 3.x.

Although we expect all functionality to work normally, there may be cases that any third-party software running on same machine as Campaign server may not yet support OpenSSL 3.x.

In such cases it may be necessary to have installed Campaign version to switch to OpenSSL 1.x.

Unica Platform documentation and help

Unica Platform provides documentation and help for users, administrators, and developers.

Table 1. Get up and running

Task	Documentation
View a list of new features, known issues, and workarounds	<i>Unica Platform Release Notes®</i>
Learn about the structure of the Unica Platform database	<i>Unica Platform System Tables Guide</i>
Install or upgrade Unica Platform and deploy the Unica Platform web application	One of the following guides: <ul style="list-style-type: none"> • <i>Unica Platform Installation Guide</i> • <i>Unica Platform Upgrade Guide</i>
Implement the Cognos® reports provided with Unica	Unica Reports Installation and Configuration Guide

Table 2. Configure and use Unica Platform

Task	Documentation
<ul style="list-style-type: none"> • Adjust configuration and security settings for products • Integrate with external systems such as LDAP and web access control • Implement single sign-on with diverse applications using SAML 2.0-based federated authentication or single sign-on • Run utilities to perform maintenance on products • Configure and use audit event tracking • Schedule runs of Unica objects 	<i>Unica Platform Administrator's Guide</i>

New Navigation UI

Unica has introduced New UI.

1. Administrators will have access to switch it for any individual user through **User settings > Edit preferences**.
2. Toggle button present at the top of the screen will help user to switch between Classic UI (Old UI) and Neo UI (New UI).

We have introduced new UI for Unica. In new UI we have Marketing Central as new home page that has got new Categorization of Unica modules bringing different modules into theme-based sections as following

- Plan & Manage
- Build & Execute
- Analyze & Optimize

Each of these categories have several tiles. Hover on each tile will display a brief on its functions. At the top left side of the screen, KitKat menu is present. Clicking on it, user can see all the options available on the UI under different sections. Also, there is link for navigating to Marketing Central page.

Plan & Manage

This section enables users to plan the marketing activities:

- Planning Workspace – This helps user to manage Plans, Programs, Projects, and Calendars.
 - Plans
 - Programs
 - Projects
 - Calendar
- Planning Workstreams - This helps users to manage Tasks, Approvals and the checklists.
 - Tasks
 - Approvals
 - Request on approval
 - Checklist
- Resource central – This helps to manage resources/Teams
- Asset Central – This helps to manage assets
- Budget Central – Track and manage accounts and invoices
- Local Campaign Subscriptions – Collaborate with fields marketer on corporate campaign
- Local Market Calendar – Collaborate calendars to view local campaigns
- Audience Central – Create and manage Audience levels
- Segment Central – Create and manage segments
- Offer Central – Create and manage offers
- Contact Central - This helps to Manage the communication preferences.

Left hand navigation will have menus specific to the accessed product from respective tile.



Note: Custom menu tiles does not show description on tool tip.

Build & Execute

This section enables user to build and further execute the planned marketing activities:

- Launch a Campaign – Helps to build and execute campaigns
- Build Strategic and Smart Segments – Helps to create reusable target audiences through campaign sessions
- Setup a Contact Strategy – Helps to optimize customer outreach
- Craft Digital Communications – Helps to create an outbound communication
 - Messaging
 - Advanced Message Editor
 - Quick Builder
- Personalize Real-time Channel Experience – Helps to create a real-time personalization strategy for your interactive channels
- Setup AI/ML for personalization – Helps to setup Learning Bins for recommendations
- Local Marketing List - Helps to manage marketing target audience list for local campaigns
- Build On-demand Campaign – Helps to view and manage on-demand campaigns
- Build a Corporate Campaign – Helps to view and manage corporate campaigns
- Orchestrate a Journey – Helps to personalize goal-based customer engagements via Journey

Analyze & Optimize

This section enables users to analyze their marketing efforts, it will optimize the marketing activities based on the analysis.

- Planning Insights - Help users to view and analyze planning
- Real-time Personalization Insights - Help users to view and analyze real-time personalization
- Campaign Insights – Help users to view and analyze campaigns
- Marketing Platform Insights - Help users to view and analyze platform
- Local Marketing Insights - Help users to view and analyze local marketing
- Messaging Insights - Help users to view and analyze messaging
- Open Insights - Help users to view and analyze through external Business Intelligence tool

Points to consider

- **Recent** menu button will maintain separate lists for Neo and Classic UI.
- When you switch from Classic UI to Neo using toggle button, a popup is invoked with confirmation message. On click of switch to Neo UI, user will be logged out. Same is true for vice-versa i.e. switching from Neo UI to Classic UI.
- Field mapping Classic UI and Neo UI

Below table shows the mapping of all menus in classic Vs Neo UI. In classic UI, we had all the menus grouped with respect to their product. In Neo UI, all those menus are grouped based on the marketing activities performed by the menu.

For example, **Request a project** menu was under **Plan** main menu in classic UI. Now it is a part of **Planning Workspace** tile present under **Plan & Manage** section. It is also renamed as **Request - Projects**.

Classic		Neo		
Plan->	Plans	Plan & Manage	Plans	
	Programs		Programs	
	Projects		Projects	
	Request a project		Request - Projects	
	Calander		Calander	
	Tasks		Tasks	
	Approvals		Approvals	
	Request an approval		Request - Approvals	
	Checklist		Checklist	
	Teams		Resource Central	
Assets	Assets Central			
Financials->	Accounts	Plan & Manage	Accounts	
	Invoices		Invoices	
Local Marketing ->	Calendar	Plan & Manage	Local Marketing Calendar	
	Subscriptions		Local Campaign Subscriptions	
Marketing Central ->	Audience Central	Plan & Manage	Audience Central	
	Segment Central		Segment Central	
	Offer Central		Offer Central	
	Contact Central		Contact Central	
Campaign ->	Campaign	Build & Execute	Launch a Campaign	
	Sessions		Build Strategic & Smart Segments	
	Optimize Sessions		Setup a Contact Strategy	
	Messaging		Craft Digital Communications ->	Messaging
	Message editor			Advanced Message Editor
	Quick Builder		Quick Builder	
Interact ->	Interactive Channels	Build & Execute	Interactive Channels	
	Global Definitions		Global Definitions (All real time attributes)	
	Global Learning		Setup AI/ML for personalization	
Local Marketing ->	Lists	Build & Execute	Local Marketing List	
	On-Demand campaigns		Build On-demand Campaign	
	Corporate campaigns		Build a Corporate Campaign	
Journey ->	Journeys	Build & Execute	Orchestrate a Journey	
Analytics ->	Plan analytics	Analyze & Optimize	Planning Insights	
	Interact analytics		Realtime Personalization Insights	
	Collaborate analytics		Local Marketing Insights	
	Platform		Marketing Platform Insights	
	Message analytics		Messaging Insights	
	Open Insights		Open Insights	
	Campaign analytics		Campaign Insights	Campaign analytics
Campaign ->	Monitoring	Build & Execute	Monitoring	

Favorites

From v12.1.5 onwards, user can add different Unica pages as Favorites for quick access. At the top of Marketing Central page, **Favorites** options is available displaying the list of favorites user has added, along with the home page button. On Unica page, at the top right corner, user can see a Star icon. On click of this, a pop up window is invoked where user can add, update or delete favorites. User can give desired name while adding the new favorites. In case the current page is already added into favorites, then the star icon is shown in Orange color and user can click on same to remove current page from favorites. Maximum 15 favorites can be added. Favorites popup can also be quickly accessed from left side navigation bar.

Set Home Page

From V12.1.5 onwards, user can set Unica page as a home page (the first page you see when you log in). Under settings, an option **Set current page as home page** is available. User can return to that page at any time by clicking the Home Page menu under Favorites.

If user has not set the homepage and they clicks on the home icon, then they will be landing on the Marketing Central landing page.

What's New

A new sub tab **What's New** is introduced under **Help**.

On click of this option user will be redirected to HCL Software website with details/features added specific to current release/version. This url can be configured under **Settings > Configuration > Unica Platform > What's New Url**.

Licensing - overview

HCL Unica products are license based and the users require to configure required licenses with the products to start using them.

The following is the list of the Unica products for which a license is mandatory:

- Unica Platform
- Unica Campaign
- Unica Interact
- Unica Deliver
- Unica Journey

After users perform clean installation or upgrade of 12.1 version of Unica products and deploy Unica products, users must configure the license. While user visits the Unica Platform application URL, it redirects to the License Details screen. Users must configure the licenses to get started with the Unica products. Only after providing valid license information users will be redirected to Unica Platform login screen.

Licensing portal

The license portal provides both Software distribution and management of your Software entitlements purchased from HCL Products and Platforms. The portal provides you with control and flexibility on how to consume your licenses. It also allows to register the licenses. Typically, an organization will have someone identified as a License Manager that has familiarity with the language of Licenses, and you may wish to add them as a user. If not, you will find these instructions sufficient to begin using your HCL Software.

Licenses and Consumption details

Users can check the licenses consumption details from HCL License Portal as well as from the Unica Platform Licensing Details page by navigating to Settings > Licensing Details page. Clicking on View License details page displays the license consumption count for all entitled products. In case there is no connectivity with License server, license consumption can be downloaded and shared with Unica.

Product Name	HCL Unica product name for which entitlement is allotted
License Type	Term/Perpetual
Start Date	Entitlement start date
Expiry Date	Entitlement expiry date (Not applicable for Perpetual license)
Entitlements available	Total number of entitlement allotted for a device or server.
Entitlements consumed	Number of entitlements consumed till now
Overdraft entitlements	Licensing model used, current model support unlimited overdrafts. (Not applicable for Perpetual license)
Overdraft consumed	Different between entitlements available entitlements consumed. (Not applicable for Perpetual license)

For more details on licensing, see the HCL Unica Licensing Guide.

License configuration

Users require to configure the license with HCL Unica products before start using it. When users access the Unica Platform login URL, it they are redirected to the License Configuration page. Users must configure the license details on this page. Unica Platform validates the license and on successful license configuration, users are redirected to the Unica Platform login screen.

License server	License Server API url, User can get the license server url from the HCL License portal.
User	HCL License Server – For any device created by default “admin” user is supported.
Password	Password set for the device
Unica Environment Type	User can specify if this is “Production” or “Non-Production” environment.
Proxy	Use proxy server to connect to HCL License Portal. Use proxy server if you do not have outbound access to HCL License Portal.
Proxy Host	Proxy server hostname or IP address
Proxy Port	Proxy server port

Proxy User	Proxy server user
Proxy Password	Proxy server user's password

All these license server details are stored in Unica Platform. User can navigate to Settings > Licensing details page if license details needs to be changed.

Licensing server availability

HCL Unica products require to be always connected to the HCL License portal. HCL Unica products become inaccessible if there is no connectivity for 5 hours between HCL Unica products and HCL License Portal server. Users are redirected to the License Details page. Once connectivity between HCL Unica and HCL License server is established users are able to access the application. HCL Unica products keep on updating the consumption details to HCL License portal every 10 mins. In case of connectivity issue, the consumption details helps the Unica Platform and once the connectivity is established the consumption is updated on the HCL License portal.

License consumption details

Users can check the licenses consumption details from HCL License Portal as well as from the Unica Platform Licensing Details page. User can navigate to Settings > Licensing Details page. Clicking on View License details page displays the license consumption count for all entitled products.

Product Name	HCL Unica product name for which entitlement is allotted
License Type	Term/Perpetual
Start Date	Entitlement start date
Expiry Date	Entitlement expiry date (Not applicable for Perpetual license)
Entitlements available	Total number of entitlement allotted for a device or server.
Entitlements consumed	Number of entitlements consumed till now
Overdraft entitlements	Licensing model used, current model support unlimited overdrafts. (Not applicable for Perpetual license)
Overdraft consumed	Different between entitlements available entitlements consumed. (Not applicable for Perpetual license)

License details page displays message indicating "No licenses configured on this environment as this environment is non-production environment". When you have selected environment as non-production environment while specifying license details.

If you require to make any non-production environment as production environment. You can change the license details and mark the environment type as "production". When you mark this environment as production environment it requires to enter all license details.



Note: License details page only displays HCL Unica products active entitlements.

For more details, see the Unica Licensing document.

Unica user account management

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

External accounts are managed in the external system.

Types of user accounts: internal and external

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



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

Identifying internal and external users in the Unica interface

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

- Internal - 
- External - 

Properties of internal user accounts

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

When a user forgets a password

Unica 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 Unica administrator can also reset any user password as needed.

Password expiration dates

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

System status of user accounts

The system status of a user is either active or disabled. A user with a disabled account cannot log in to any Unica 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 Unica Campaign listener runs as root on a UNIX™-type system.

Type of user

User can be of type Full or Lite. Full user is regular Platform user. By default, user type will be Full.

Data sources

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

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

Adding internal user accounts

Use this procedure to add internal user accounts.

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

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

4. Click **OK**.

The new user name appears in the list.

Deleting internal user accounts

Use this procedure to delete internal user accounts.



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

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

Changing internal user password expiration dates

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



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

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

Resetting internal user passwords

Use this procedure to reset internal user passwords.

1. Click **Settings > Users**.

The **Users** list is displayed in the left pane.

2. Click the user name you want to change.
3. Click the **Reset password** link at the bottom of the page.

4. Enter the new password in the **Password** field.
5. Enter the same password in the **Confirm** field.
6. Click **Save changes** to save your changes.
7. Click **OK**.



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

Changing internal user account properties

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

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

Changing internal user system status

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

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



Note: If you select **DISABLED**, the user will no longer be able to log in to any Unica applications. Users with Admin access to Unica Platform cannot disable themselves.

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

Changing internal user type

Use this procedure to change the type of internal users:

1. Click **Settings > Users**
2. Click the name of the account you want to change.
3. Click on the **Edit properties** link at the bottom of the page.
4. Select the type in the Type drop-down list. The options are FULL and LITE.



Note: Dropdown will be shown as read only if active license for **Unique_Platform_Lite_Users** entitlement does not present.

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

Adding internal user data sources

Use this procedure to add data sources for internal users.

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

Changing internal user data sources

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

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

If you do not set a new password, the old one is retained.

6. Complete the form and click **Save changes** to save your changes.
7. Click **OK**.

Deleting internal user data sources

Use this procedure to delete internal user data sources.

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

The user management pages

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

The New user page

Table 3. Fields on the New user page

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 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, <code>password</code> is not the same as <code>Pass-word</code>. • You may use any character when you create or reset a password in Unica. <p>Additional password requirements are set on the Configuration page. To see what they are for your installation of Unica, 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.
Company	The user's company.
Country	The user's country.
Address	The user's address.
Work phone	The user's work phone number.

Table 3. Fields on the New user page (continued)

Field	Description
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 Unica Campaignlistener runs as root on a UNIX™-type system.
Status	Select ACTIVE or DISABLED from the drop-down list. ACTIVE is selected by default. Disabled users are prevented from logging in to all Unica applications.
Type	Select FULL or LITE from the drop-down list. By default, Full is selected.

The Edit properties page

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

Table 4. Fields on the Edit properties page

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

The Reset password page

Table 5. Fields on the Reset password page

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

The New Data Source and Edit Data Source Properties pages

Table 6. Fields on the Data Source pages

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

Locale preference

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

There are two ways to set locale in Unica Platform.

Globally

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

Per user

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

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



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

Setting the user locale preference

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

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

Synchronization of external users

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

Automatic synchronization has limited functionality.

- Users deleted from the LDAP server are not deleted during automatic synchronization.

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

Forcing synchronization of external users

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

1. Log in to Unica and click **Settings > Users**.
2. Click **Synchronize**.

Users and groups are synchronized.

Security management

Unica Platform supports roles and permissions to control user access to objects and features in Unica applications.

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

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

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

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

Permissions and tasks of the security administrator in Unica Platform

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

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

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

- Create internal groups and manage their memberships and partition assignments.
- Create roles for Unica Platform and Unica Campaign, if necessary, and assign permissions to these roles.
- Manage user access to Unica 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 Unica Platform

Special characters in role and policy names

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

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

Roles and permissions in Unica Platform and Unica Campaign

Roles in Unica Platform and Unica Campaign are a configurable collection of permissions. For each role in Unica Platform and Unica 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 Unica. 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 Unica Campaign, the ability to perform a function on a particular object is granted or denied based on the security policy of the object.

Overview of managing user application access in Unica Platform

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

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

Types of groups: internal and external

When Unica 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 Unica using the security user interface. These users are authenticated through Unica.
- **External** - Unica groups that are mapped to groups in the external system. Examples of external servers are LDAP and web access control servers.



Attention: A group referred to as an external group in this guide is one that is actually created in Unica 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 Unica 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 Unica 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.

Partitions and security management

Partitions in Unica 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 Unica 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 Unica Campaign installation documentation.

Pre-configured users and roles

When Unica is first installed, three users are pre-configured and are assigned system-defined roles in Unica Platform and Unica 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 Unica administrator to manage product configuration, users, and groups across all partitions in a multi-partition environment, and to use all Unica Platform features (except reporting, which has its own roles) without any filtering by partition. By default, this account has the following roles in Unica Platform.

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

These roles allow the platform_admin user to perform all administrative tasks within Unica 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 Unica Campaign, in the default partition, partition1
 - The Global policy Admin role

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

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

The asm_admin user account

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

- In Unica 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 Unica 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 Unica 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 Unica 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 Unica Platform. By default, this user does not have access to any of the Unica products.

- In Unica 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 Unica Campaign.

Cross-partition administration privileges

In a multi-partition environment, at least one user account with the `PlatformAdminRole` role in Unica Platform is required, to enable you to administer security for Unica 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 Unica. 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 Unica Platform, log in as that user and reset the `platform_admin` user's password or create another account with the `PlatformAdminRole` role in Unica Platform.
- If you have only one user with the `PlatformAdminRole` role in Unica Platform (for example, `platform_admin`), and this user is disabled, you can create a new `platform_admin` account using the `restoreAccess` utility provided with Unica Platform.

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

Adding an internal group

Use this procedure to add an internal group.

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

 **Important:** Do not give the group a the same name as system-defined roles. For example, do not name a group "Admin," which is a role name used in Unica 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 an internal subgroup.

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

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

5. Click **Save changes**.

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

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

Deleting a group or subgroup

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

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



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

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

Changing a group or subgroup description

Use this procedure to change a group or subgroup description.

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



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

3. Click **Edit properties**.
4. Edit the description as desired.
5. Click **Save changes** to save your changes.
6. Click **OK**.

Assigning a group to a partition

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

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

This field is available only when multiple partitions are configured.

6. Click **Save changes** to save your changes.
7. Click **OK**.

Adding a user to a group or subgroup

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

1. Click **Settings > Users**.



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

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

The group name moves to the **Groups** box.

6. Click **Save changes** to save your changes.
7. Click **OK**.

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

Removing a user from a group or subgroup

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



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

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

The group name moves to the **Available groups** box.

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

The user group management pages

These are the fields you use to configure user groups.

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

Table 7. Fields on the New group, New subgroup, and Edit properties pages

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 Unica Campaign. Doing so can cause problems during upgrades.</p> <p>Unica names preserve case for display purposes, but use case-insensitive rules for comparison and creation (for example, you cannot create both Admin and admin as 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>
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>

Fields on the Edit users and Edit roles pages

Table 8. Fields on the Edit users and Edit roles pages

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

Creating a role

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

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

Modifying role permissions

Use this procedure to modify role permissions.

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

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

Removing a role from the system

Use this procedure to remove a role from Unica.



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

1. Click **Settings > User roles & permissions**.
2. Click the plus sign next to a product in the list on the left, and then click the name of the partition where you want to create the role.
3. For Unica 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.

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.

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

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

Roles that are not assigned to the user are shown in the **Available Roles** box on the left. Roles that are currently assigned to the user are shown in the **Selected roles** box on the right.

4. Select a role in the **Available roles** box. Complete one of the following tasks:
 - To assign a role to a user, select a role in the **Available roles** box, and click **Add**.
 - To remove a role from a user, select a role in the **Selected roles** box, and click **Remove**.
5. Click **Save changes**, and then click **OK**.

Definitions of permission states

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

These states have the following meanings.

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

Permissions for products that use only basic roles

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

Table 9. Permissions for products that use only basic roles

Application	Roles
Leads	Leads roles are reserved for future use.
Reports	<ul style="list-style-type: none"> • ReportsSystem - grants the <code>report_system</code> permission, which gives you access to the Report SQL Generator and Sync Report Folder Permissions options in the Settings menu. • ReportsUser - grants the <code>report_user</code> permission, which is used by the Authentication Provider installed on the IBM® Cognos® 11 BI system only. <p>For information about authentication options for the IBM® Cognos® 11 BI integration and how the Authentication Provider uses the reporting permissions, see the Cognos Reports Installation and Configuration Guide.</p>
Unica Deliver	<ul style="list-style-type: none"> • Deliver_Admin - Has full access to all features. • Deliver_User - Reserved for future use. <p>Access is further defined through the security policies in Unica Campaign. See the Unica Deliver Startup and Administrator's Guide for details.</p>

Table 9. Permissions for products that use only basic roles (continued)

Application	Roles
Unica Interact	<ul style="list-style-type: none"> • InteractAdminRole - Has full access to all features.
Unica Collaborate	<ul style="list-style-type: none"> • collab_admin - Has full access to all features. • corporate - Can use Unica Campaign and Unica Collaborate 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 Unica Collaborate.
Unica Plan	<ul style="list-style-type: none"> • PlanUserRole - By default, users with the PlanUserRole role have very few permissions enabled in Unica Plan. 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 Unica Plan, including access to all administrative and configuration settings, allowing a broad range of access. <p>Access is further defined through the security policies in Unica Plan.</p>
Unica Journey	<ul style="list-style-type: none"> • Journeyadmin: Users have access to all administrative and configuration settings, allowing a broad range of access. • Journeyuser: Users have limited access to administrative and configuration settings. They can only view the settings but cant perform crude operations on them.
Unica Centralized Offer Management	<p>OfferAdmin: Users have access to all administrative and configuration settings, allowing a broad range of access.</p> <p>OfferUser: Users have limited access to administrative and configuration settings.</p>

Table 9. Permissions for products that use only basic roles (continued)

Application	Roles
IBM SPSS Modeler Advantage Enterprise Marketing Management 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 Enterprise Marketing Management Edition, including access to all administrative and configuration settings.

Permissions for Unica Platform

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

Table 10. Unica Platform permissions

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

Table 10. Unica Platform permissions (continued)

Permission	Description
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.

Permissions for Opportunity Detect

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

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

Table 11. Permissions in Opportunity Detect

Permission	Description
View only	Can access all of the user interface, in view-only mode.
Design triggers	<ul style="list-style-type: none"> • Can create workspaces and design trigger systems. • Can create, modify, and delete all trigger related resources. • Can access Workspace, Component, Audience Level, Data Source, and Named Value List pages. • Can not access the Server Groups page or the Deployment tab of a workspace. • Can not set off a batch run. • Can not administer objects that the web service creates when Opportunity Detect is integrated with Unica Interact.
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 page and the Deployment tab of a workspace, but can not designate a server group for production. • Can not deploy deployment configurations or run deployment configurations that use a production server group.
Run for production	<ul style="list-style-type: none"> • Deploy deployment configurations and run batch deployment configurations on any server group. • Perform all actions on the Server Group page and the Deployment and Batch Run tabs of a workspace, including designating a server group for production.
Administer real time	Manage objects that the web service creates when Opportunity Detect is integrated with Unica Interact to enable real time mode.

Table 11. Permissions in Opportunity Detect (continued)

Permission	Description
	<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.

Configuration management

When Unica is first installed, the Configuration page shows only the properties used to configure Unica Platform and some global configuration properties. When you install additional Unica applications, the properties used to configure these applications are registered with Unica 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.

Property categories

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




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

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

Category types

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

Table 12. Icons for category types

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

Templates for duplicating categories

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

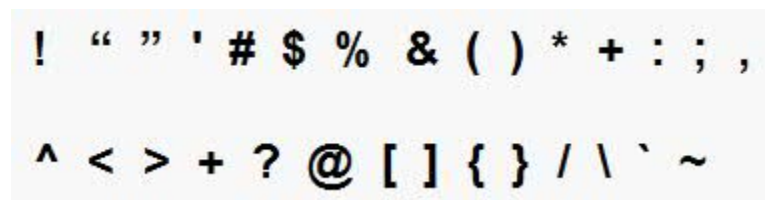
To create a category, you duplicate the template. For example, you can create a new Unica Campaign partition or data source by duplicating the appropriate template.

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

Category naming restrictions

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

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




Also, the name cannot start with a period.

Property descriptions

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

- Click **Help > Help for this page** to launch online help and display a topic that describes all of the properties for the page you are viewing.
- Click **Help > Product documentation** to launch a page that gives you access to all of the product documentation in online or PDF format. All property descriptions are included as an appendix in the Unica Platform Administrator's Guide.

The refresh function

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

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



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

The default user locale preference

Unica Platform contains a default locale attribute that applies to all Unica applications that implement it.

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

For details on this property, see its online help in the Configuration area or the Unica Platform Administrator's Guide.

To learn whether an Unica 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.

Navigating to a category

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

1. Log in to Unica.
2. Click **Settings > Configuration** in the toolbar.

The Configuration page shows the Configuration Categories tree.

3. Click the plus sign beside a category.

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

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

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

Editing property values

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

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

The Settings page for the category shows a list of all the properties in the category and their current values.

2. Click **Edit settings**.

The Edit settings page for the category shows the property values in editable fields.

3. Enter or edit values as needed.

In UNIX™, all file and directory names are case-sensitive. The case of any file and folder name you enter must match the case of the file or folder name on the UNIX™ machine.

4. Click **Save changes** to save your changes or **Cancel** to exit the page without saving.

Creating a category from a template

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

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

Unlike other categories, template category labels are in italics and enclosed in parentheses.

2. Click the template category.
3. Enter a name in the **New category name** field (required).
4. You can edit properties within the new category now, or later.
5. Click **Save changes** to save the new configuration.

The new category appears in the navigation tree.

Deleting a category

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

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

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

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

2. Click the **Delete category** link.

A window shows the message, Are you sure you want to delete "category name"?

3. Click **OK**.

The category no longer appears in the navigation tree.

Dashboard management

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

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

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

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 Unica 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 Unica Campaign, you may want to include the My Custom Bookmarks portlet, one of the pre-defined Unica portlets.

User permissions required to view dashboards

Dashboards allow Unica users to view pages from multiple products (such as Unica Plan and Unica 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 Unica 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

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

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

- List: A list of Unica items specific to the user. Examples of list portlets are My Recent Campaigns (Unica Campaign), My Alerts (Unica Plan), and the Continent Summary report (Digital Analytics for On Premises).
- IBM® Cognos® or Unica Insights reports: A specially formatted version of Unica reports.

You can also create your own custom dashboard portlets.

Pre-defined portlet availability

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

You must enable the pre-defined portlets in Unica Platform before you can use them in a dashboard. Unica portlets are listed in Unica 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.

Unica Plan report portlets

The following table describes the Unica Plan dashboard portlets that are available after the Unica Insights is installed or the Cognos Unica Plan Reports package for Cognos is installed.

Table 13. Standard Unica Plan report portlets

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

Unica Plan list portlets

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

Your system administrator selects the portlets that members of your organization can add to the dashboard. To manage your dashboards and add portlets to them, select **Dashboard > Create Dashboard**.

Table 14. Standard Unica Plan list portlets




Report	Description
Approvals Awaiting Action	List of approvals that wait 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 Unica Plan alerts.
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> $\frac{\text{Number of Finished Tasks} + \text{Number of Skipped Tasks}}{\text{Total Number of Workflow Tasks}}$ <ul style="list-style-type: none"> • To recalculate project health status, click . The system recalculates the health status for display by this portlet only. It does not work elsewhere in Unica Plan. <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. • You can view the summary information for a project on the Summary tab. To view more metrics for project health, click the percentage complete indicator. To view the My Tasks list, click the number in the Tasks column.
My Requests	Lists requests that you own.
My Tasks	Lists tasks that you own.
Projects Over Budget	Lists all projects that are over budget for the calendar year.

Table 14. Standard Unica Plan list portlets (continued)

Report	Description
	 Note: This report requires the Financial Management module.

Report portlets for Unica Campaign

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

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

Table 15. IBM® Cognos® report portlets for Unica Campaign

Report	Description
Unica Campaign Return on Investment Comparison	A report that compares, at a high level, the ROI of campaigns created or updated by the user viewing the report.
Unica Campaign Response Rate Comparison	A report that compares the response rates of one or more campaigns created or updated by the user viewing the report.
Unica Campaign Revenue Comparison by Offer	A 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	A 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	A report that compares the response rate by offer created or updated by the user viewing the report.
Offer Response Breakout	A report that shows the active offers created or updated by the user viewing the report, broken out by status.

Unica Campaign list portlets

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

Table 16. Unica Campaign list portlets

Report	Description
My Custom Bookmarks	A list of links to websites or files created by the user viewing the report.
My Recent Campaigns	A list of the most recent campaigns created by the user viewing the report.

Table 16. Unica Campaign list portlets (continued)

Report	Description
My Recent Sessions	A list of the most recent sessions created by the user viewing the report.
Unica Campaign Monitor Portlet	A list of the campaigns that have run or are currently running that were created by the user viewing the report.

Unica Deliver report portlets

The following dashboard portlets are available with unica Insights or with the Unica Deliver 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.

Unica Interact report portlet

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

This dashboard report is defined to point to the interactive channel with the ID of 1. To create additional versions of this report (to report on additional interactive channels) or to change the ID of the interactive channel that this report points to, see [How to configure the Interaction Point Performance dashboard portlet \(on page 52\)](#).

How to configure the Interaction Point Performance dashboard portlet

Unica Interact has one 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 Cognos® reports, you need Cognos® report authoring skills. For detailed documentation about creating and editing Cognos® BI reports, see the Cognos® BI documentation, especially Cognos® BI Report Studio Professional Authoring User Guide for your version of Cognos®.

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

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

Unica Collaborate list portlets

This section describes the standard Unica Collaborate portlets that are available for use on dashboards.

Table 17. Unica Collaborate list portlets

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

Unica Optimize list portlets

The standard Unica Optimize portlets that are available for use on dashboards.

These portlets are available for use in the Unica dashboard only.

Table 18. Unica Optimize list portlets

A two-column table that describes the Unica Optimize list portlets.

Report	Description
My recent Unica Optimize sessions	A list of the last 10 Unica Optimize sessions run by the user viewing the report within the last 30 days.
My recently successful Unica Optimize run instances	A list of the last 10 Unica Optimize sessions run by the user viewing the report that completed successfully within the last 30 days.
My recently failed Unica Optimize run instances	A list of the last 10 Unica Optimize sessions run by the user viewing the report that did not complete successfully within the last 30 days.

Pre-assembled dashboards

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

Pre-assembled dashboard availability

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

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

List of pre-assembled dashboards

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

Table 19. List of pre-assembled dashboards

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"> • Unica Campaign • Unica Insights or Unica 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"> • Unica Plan • Unica Insights or Unica Plan 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 	Unica Plan

Table 19. List of pre-assembled dashboards (continued)

Pre-assembled dashboard	Purpose	Portlets	Required products
Project Requests and Approvals	This dashboard shows tasks that require action, and provides status updates on projects and a high level overview of the marketing financial position and where funds are being spent.	<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"> • Unica Plan with the Financial Management Module • Unica Insights or Unica Plan 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"> • Unica Plan with the Financial Management Module • Unica Insights or Unica Plan 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 Unica 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 Unica Campaign pre-defined portlets
- The Unica Plan 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 \(on page 56\)](#).

Data considerations

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

User expectations

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

Guidelines

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

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

Scheduling a dashboard report

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



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

1. In Cognos®, copy the report and save it under a new name.
2. In Cognos®, open the copied report and save it as a view with the same name as the original report. Save it in the `Unica Dashboard/Product` folder, where `Product` is the appropriate product folder

3. In Cognos®, schedule the view.
4. In Unica, add the report to the dashboard, if you have not done so already.
5. Only if the report is one of the pre-defined portlets, do the following in Unica.
 - 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**.

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 Unica Platform.

When Unica 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 Unica Platform can assign any Unica user to administer individual dashboards in the partition to which that user belongs. Dashboard administration is done in the dashboard administration area of Unica Platform.

Dashboard layout

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

The options are as follows.

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

Dashboards and partitions

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

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

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

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

Overview of working with dashboards in a multi-partition environment

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

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

Only the `platform_admin` user, or another user with the PlatformAdminRole permissions can perform this task.

2. For each partition, ensure that at least one user has the Administer Dashboards permission, and make a note of these user names.

The Unica 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 Unica.

Refer to the list of users you created in the previous step.

- b. On the **Settings > Dashboard portlets** page, enable pre-defined portlets as needed.
- c. On the Dashboard Administration page, create the needed dashboards and add portlets.
- d. For each non-global dashboard, assign users who can view the dashboard.

You can assign individual users or groups to the dashboard.

- e. For each dashboard, assign one or more users as dashboard administrator.

Enabling or disabling pre-defined portlets

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

1. Log in to Unica 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

1. In Unica, select **Dashboard** to open the Dashboard administration page.

All dashboards associated with your partition are shown.

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

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

You can now add portlets.

Creating a pre-assembled dashboard

Use this procedure to create a pre-assembled dashboard.

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

The available pre-assembled dashboards are listed.

5. Select the pre-assembled dashboard you want to use and click **Next**.

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

6. Click **Save** to finish creating the dashboard.

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

Adding a pre-defined portlet to a dashboard

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

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

You can also access the Manage portlets page from the Administration tab, by clicking the Manage Portlets icon on the dashboard.

3. Select the check box next to one or more portlets to select it for addition to the dashboard.

Use the following features to assist you in selecting portlets.

- Filter the list of portlets by name or by the product that is the source of the portlet.
- Display all portlets at once or page through the list.
- Click column headings to sort the list alphabetically by source or portlet name, in ascending or descending order.

4. Click **Update**.

The selected portlets are added to the dashboard.

Removing a portlet from a dashboard

Use this procedure to remove a portlet from a dashboard.

1. In Unica, select **Dashboard**.

A Dashboard Administration page opens. All dashboards associated with your partition are shown, with their portlets listed.

2. In the dashboard where you want to remove a portlet, click the **Delete** icon next to the portlet you want to remove.
3. Click **Yes, Delete** at the prompt.

The portlet is removed from the dashboard.

Changing the name or properties of a portlet

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

1. In Unica, select **Dashboard**

A Dashboard Administration page opens. All dashboards associated with your partition are shown, with their portlets listed.

2. In the dashboard you want to work with, click the **Edit Portlet** icon next to the portlet whose name you want to change.

An Edit Portlet window opens.

3. Edit the name, description, URL, or hidden variables of the portlet.
4. Click **Save**.

Changing the name or properties of a dashboard

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

1. In Unica, select **Dashboard**

A Dashboard Administration page opens. All dashboards associated with your partition are shown.

2. In the dashboard you want to work with, click the **Manage Settings** icon at the bottom of the dashboard.

A Settings tab opens.

3. Click the **Edit Dashboard** icon.

An Edit Dashboard window opens.

4. Edit the title, description, or type of the dashboard, enable or disable it, or change whether users can change the layout..

5. Click **Save**.

Deleting a dashboard

Use this procedure to delete a dashboard.

1. In Unica, select **Dashboard**

A Dashboard Administration page opens. All dashboards associated with your partition are shown.

2. In the dashboard you want to work with, click the **Delete Dashboard** icon at the bottom of the dashboard.

3. When prompted, click **Yes, Delete**.

The dashboard is deleted.

Assigning or changing a dashboard administrator

Use this procedure to assign or change a dashboard administrator.

1. In Unica, 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**.

The Manage Portlets page

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

Table 20. Fields on the Manage Portlets page

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

Quick link portlets

Quick links are pre-defined links to Unica products. Some quick links enable users to perform basic actions in the Unica 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 Unica products are installed when the product is installed. As of the 9.0.0 release, only Unica Plan provides quick links. The same security considerations apply for quick links as for pre-defined portlets.

To add a quick links portlet to one of your dashboards, click **Manage Portlets > Create Quick Link Portlet** and select the quick links you want to include.

The following table describes the quick links available when Unica Plan is installed.

Table 21. List of quick link portlets

Quicklink	Function
Create New Project Request	Opens up a popup window where you can choose a project template to create a Project Request. You can also click Continue to open the Project Request wizard in the application.
Create New Project	Opens up a popup window where you can choose a Project template to create a Project. You can also click Continue to open the Project wizard in the application.

Table 21. List of quick link portlets (continued)

Quicklink	Function
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.

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 Unica pages.

- Any Cognos® report, including Unica 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 *Unica Reports Installation and Configuration Guide*.
- Quick links portlets, which you can build using pre-defined links to Unica 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 Unica Platform for authentication or to use no authentication, or a dashboard report from any other Unica product that uses Unica 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 Unica Platform for authentication, the user must enter login credentials one time per browser session.
- If your portlet is a NetInsight OnDemand report or an internet or intranet page that requires authentication, the portlet behaves as a browser would. The user must enter login credentials in the content of the page the first time they view it during a browser session, and cookies are used to keep the user logged in.
- If your portlet is an IBM Digital Analytics report, users can view only those reports for which they have permissions in Digital Analytics. Also, if single-sign-on is enabled with Digital Analytics, users can view Digital Analytics reports in Unica Platform dashboards without entering their credentials. Otherwise, users must enter their Digital Analytics credentials to view Digital Analytics reports in Unica Platform dashboards.

Overview of the portlet creation process

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

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

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

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

You can create portlets from the following sources.

- Digital Analytics for On Premises report
- IBM Cognos® report
- Digital Analytics report
- NetInsight OnDemand report and pages on the internet or your company intranet

2. Add the URL to the `Platform_Admin_URL.properties` file.

The `Platform_Admin_URL.properties` file is located in the `conf` directory under your Unica Platform installation.


3. Stop and restart the Unica Platform web application.
4. Add the portlet to a dashboard.

Preparing the URL from a Digital Analytics for On Premises report

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

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

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

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

The Export options window opens.

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

The report URL is displayed in a dialog box.

5. Copy the URL and paste it into a text editor.
6. Add the following to the beginning of the report URL:

`Your_HCL_Unica_URL/suiteSignOn?target=`

where `Your_HCL_Unica_URL` is the login URL for your installation of Unica.

For example, suppose you have the following information.

- Your report URL is `MyReportURL`
- The login URL for your installation of Unica is `http://myHost.myDomain:7001/unica`

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

Preparing the URL from an IBM® Cognos® dashboard report

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

For information about creating dashboard reports with IBM® Cognos®, see the Unica 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 Unica application's subfolder in the **Unica Dashboards** folder on the IBM® Cognos® system. That is: *Campaign*, *Interact*, or *Plan* for Unica Plan. (Plan was the previous name of the Unica Plan application.)
- *ReportName* is the name of the dashboard report. For example: Campaign Performance Comparison

For example,

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

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

```
&isView=true
```

Preparing the URL from a Digital Analytics report

Use this procedure for Digital Analytics reports.

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

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

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

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

Use the copied URL when you create your custom portlet.

Adding a custom portlet to a dashboard

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

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

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

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

A **Manage Portlets** window opens.

3. Click **Create Custom Portlet**.

A **Create Custom Portlet** window opens.

4. Do one of the following sets of steps, depending on the type of portlet you are adding.

If you are creating a portlet that is not a Digital Analytics report portlet, do the following.

- For the **Type**, select **Custom**.
- Complete the **Name** and **Description** fields.
- Paste the contents of your clipboard (which contains the URL you obtained earlier) into the **URL** field.

If you are creating a Digital Analytics report portlet, do the following.

- For the **Type**, select **IBM Digital Analytics**.
- Complete the **Name** and **Description** fields.
- Paste the contents of your clipboard (which contains the URL you obtained earlier) into the **IBM Digital Analytics URL** field.

5. Click **Save**.

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

Dynamic tokens

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

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

The following tokens are supported.

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

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

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

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

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

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

The Create Custom Portlet page

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

Table 22. Fields on the Create Custom Portlet page

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

Dashboard membership administration

Topics in this section describe how to manage dashboard membership.

The dashboard administrator

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

Granting or removing dashboard membership

Use this procedure to grant or remove dashboard membership.

1. In Unica, 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**.

The Unica Scheduler

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

Items that you can schedule

You can schedule the following.

- Unica Campaign flowchart runs



Note: The Unica Scheduler is completely independent of the Schedule process in Unica Campaign.

- Unica Optimize optimization session and post-optimization flowchart runs
- Unica Deliver mailings
- Unica Plan bulk deactivations
- Calls to external APIs
- Unica alerts and notifications
- External batch or shell scripts

Schedules and runs

The scheduler uses two basic concepts: schedules and runs.

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

Types of schedules

There are three types of schedules.

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

Schedule notifications

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

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 Unica Campaign flowchart can send a trigger that starts an Unica Deliver mailing.

A trigger is a text string that the Unica 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 Unica 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 \(on page 255\)](#) utility. You can also use triggers to cause a set of flowcharts to run in series, one after another.

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

- Flowchart 1 is scheduled with a "Flowchart 1 run complete" trigger that is sent when the run completes successfully.
- Flowchart 2 is scheduled as follows.

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

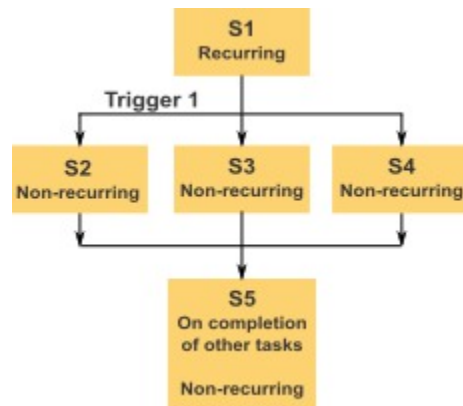
About start triggers

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

Schedules that depend on completion of multiple runs

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

For example, suppose you have a schedule, S1, that is set up with a recurrence pattern. S1 has a trigger that is sent every time an S1 run completes successfully. Three schedules, S2, S3, and S4, are configured to start when they receive the outbound trigger from S1. You can set up an additional schedule, S5, that runs when S2, S3, and S4 complete successfully. S5 runs only when all three of the runs on which it is dependent complete. The following diagram illustrates this example.



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

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

- The schedules on which the schedule you are configuring depends must be non-recurring. In the example above, S2, S3, and S4 must be non-recurring. However, because S1 recurs, S2, S3, and S4 effectively recur, based on S1 runs.
- The schedule that is dependent on other schedules must also be non-recurring. In the example, S5 must be non-recurring. Again, because S1 recurs, S5 effectively recurs as well.

- The schedule that is dependent on other schedules cannot be used as one of the criteria in the **On completion of other tasks** option for any other schedule. In the example, S5 cannot be used as a criterion in the **On completion of other tasks** option for any other schedule.
- If you want to delete a schedule that is configured with the **On completion of other tasks** option, you must first change the configuration to remove the **On completion of other tasks** option. Then you can delete the schedule.

Schedule triggers that are sent from an external script

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

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

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

The utility can provide the following information.

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

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

Example script

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

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

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

```
@rem*****
@rem This script is used to call the 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 Unica Platform is installed. However, you can select from any other time zones listed in the **Select time zone** drop down list. These options are expressed as GMT times followed by the commonly used term for that time zone. For example, (GMT-08:00) Pitcairn Islands or (GMT-08:00) Pacific Time (US & Canada).

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

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

Scheduler throttling

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

The throttling threshold is the highest number of runs associated with that group that can run concurrently. To reduce resource consumption on the server, you can set the throttling threshold to a smaller value. Only schedules created in the Unica 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 Unica Campaign or by using the Unica 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 Unica 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 Unica 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 for the Unica Scheduler

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

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

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

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

You must create a throttling group for each type of object whose runs you want to control in this way. For example, flowchart throttling groups are available only for scheduling flowcharts; mailing throttling groups are available only for scheduling mailings.

3. Assign one or more schedules to the group, as needed.

Whitelist prerequisite for external tasks (with FixPack 10.0.0.1 only)

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

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

Adding a script to the whitelist

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

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

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

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

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

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

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

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

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

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

3. Save and close the whitelist file.

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

Adding an API to the whitelist

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

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

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

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

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

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

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

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

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

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

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

3. Save and close the whitelist file.

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

Best practices for setting up schedules

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

To create a schedule wizard

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

The following table describes the fields you use when you schedule runs of Unica Campaign flowcharts, Unica Deliver mailings, Unica Optimize sessions, external scripts, and API calls.

Table 23. Fields in the create a schedule wizard

Field	Description
Select a task type	<p>The type of object to be scheduled. The following options are available.</p> <ul style="list-style-type: none"> • External task - script <p>Allows you to schedule invocation of tasks defined in batch or shell scripts external to Unica.</p> <p>Only if you have applied Unica Platform FixPack 10.0.0.1, the script must be listed in a whitelist file located in the <code>conf</code> directory under your Unica Platform installation. Also, the script must be on the web application server where Unica Platform is deployed.</p> <ul style="list-style-type: none"> • External task - API <p>Allows you to schedule invocation of APIs external to Unica.</p> <p>Only if you have applied Unica Platform FixPack 10.0.0.1, the API must be listed in a whitelist file located in the <code>conf</code> directory under your Unica Platform installation.</p>

Table 23. Fields in the create a schedule wizard (continued)

Field	Description
	<ul style="list-style-type: none"> <li data-bbox="565 317 867 344">• Unica Campaign flowchart <p data-bbox="581 380 1422 485">Allows you to schedule invocation of Unica Campaign flowcharts. Selecting this option takes you to the Unica Campaign list page where you select a campaign, optionally set flowchart override parameters, and schedule the flowchart run.</p> <li data-bbox="565 516 834 543">• Unica Optimize session <p data-bbox="581 579 1390 684">Allows you to schedule invocation of Unica Optimize sessions. Selecting this option takes you to the Unica Optimize sessions list page where you select a session and schedule the session.</p> <li data-bbox="565 716 808 743">• Unica Deliver mailing <p data-bbox="581 779 1390 884">Allows you to schedule invocation of Unica Deliver mailings. Selecting this option takes you to the Unica Deliver mailings list page where you select and schedule the mailing.</p> <li data-bbox="565 915 886 942">• Unica Plan bulk deactivation <p data-bbox="581 978 1414 1083">Allows you to schedule bulk deactivation of projects in Unica Plan. Selecting this option takes you to the Unica Plan Administrative Settings page where you click Deactivation Administration and schedule the bulk deactivation.</p> <li data-bbox="565 1115 639 1142">• Alert <p data-bbox="581 1178 1414 1283">Allows you to schedule alerts for users of Unica. Selecting this option opens a window where you set the message title, message body, and severity. After you click Schedule this alert, you can create a schedule for the alert.</p> <p data-bbox="581 1318 1305 1346">Users can manage their notification subscriptions based on severity.</p> <li data-bbox="565 1377 708 1404">• Notification <p data-bbox="581 1440 1414 1587">Allows you to schedule notifications for users of Unica. Selecting this option opens a window where you set the message title, message body, and severity. After you click Schedule this notification, you can create a schedule for the notification.</p> <p data-bbox="581 1623 1305 1650">Users can manage their notification subscriptions based on severity.</p>
Schedule name	Enter a name for the schedule.
Scheduler group	If you have created one or more throttling groups, you can associate this schedule with a group to limit the number of runs of this schedule that can execute at the same time. Throttling groups configured on the Configuration page appear as options in this field

Table 23. Fields in the create a schedule wizard (continued)

Field	Description
Description	Enter a description for the schedule.
Run parameters	<p>Used when you schedule APIs and scripts.</p> <p>Only if you have applied Unica Platform FixPack 10.0.0.1, a whitelist prerequisite applies to any external tasks that you create to schedule API calls or scripts. Before you can schedule an external task, you must add the API or script to a whitelist located in the <code>conf</code> directory under your Unica Platform installation.</p> <ul style="list-style-type: none"> For API schedules, enter the URI, plus any parameters in the format shown in the examples. <p>API with no parameters: <code>http://example.com</code></p> <p>API with parameters: <code>http://www.example.com/tickets?fields=id&state=open&sort=updated_at</code></p> <p>Currently, there is no support for Unica Platform tokens in the URI.</p> For script schedules, enter the full path to the script on the Unica Platform server, plus any parameters in the format shown in the examples. Specify the parameter names between double number signs (##) followed by a space. <ul style="list-style-type: none"> Windows™ examples <p>Script with no parameters: <code>C:\Scripts\ExecuteDatabaseJob.bat</code></p> <p>Script with parameters:</p> <p><code>C:\Scripts\RunETLJobs.bat ##username## ##password## ##db_table##</code></p> UNIX™ examples <p>Script with no parameters: <code>/opt/ExecuteDatabaseJob.sh</code></p> <p>Script with parameters:</p> <p><code>/opt/RunETLJobs.sh ##username## ##password## ##db_table##</code></p> <p>Execution of these tasks is asynchronous. Unica Platform does not track success or failure of script and API tasks. Status indicates only whether they are launched successfully.</p>
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.

Table 23. Fields in the create a schedule wizard (continued)

Field	Description
On error, send a trigger	If you want runs of this schedule to send a trigger when the run fails, enter the trigger text here. Other schedules can be set to listen for this trigger.
Search tags / keywords	Enter any tags you want you associate with the schedule for use in searches. Separate multiple entries with commas.
Schedule status	Whether the schedule is enabled or disabled. Disabling a schedule applies only to future or queued runs of that schedule. Any run currently underway is not affected. The default status is Enabled .
Select time zone	If you select an option other than the server default, the Start , End , and Last updated columns on the Schedule management page show both the server default time and the time in the selected zone.
When to start	<p>Select one of the following options to specify the first time the schedule runs. The start time applies only to the first run; it defines the time when a schedule is first eligible to run. The actual first run might be after the start date if any of the following conditions are present.</p> <ul style="list-style-type: none"> • The schedule is configured to wait for a trigger. • The schedule is a member of a throttling group. • The schedule uses a recurrence pattern. <ul style="list-style-type: none"> • Now • On a date and time - Select a date and time. • On a trigger - Select an existing trigger or enter a new one. If you enter a new one, you must configure a schedule to send this same string on success or failure. • On a trigger after a date - Select an existing trigger or enter a new one, and select a date and time. If you enter a new trigger, you must configure a schedule to send this same string on success or failure. • On completion of other tasks - Select from a list of existing schedules. The schedule runs only when the selected other schedules have finished their runs.
Number of runs	<p>Select one of the following options to specify the number of runs.</p> <ul style="list-style-type: none"> • Run once only - The schedule runs one time. It is eligible to execute the run on the start date and time you specify. • Stop after n occurrences - Runs stop after the specified number of runs have occurred (whether the runs succeed or fail) or the end date arrives, whichever is first.

Table 23. Fields in the create a schedule wizard (continued)

Field	Description
	<ul style="list-style-type: none"> • 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. <p>When you click the Set up recurrences button, you can 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 Unica 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.

Run exclusions

10.0.02

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

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

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

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

Viewing exclusion rules

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

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

To view exclusion rules, complete the following steps:

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

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

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

Adding exclusion rules

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

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

To add an exclusion rule, complete the following steps:

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

By default, **Enabled** is selected.

7. Select the **Exclusion type**.

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

- a. Select the **Time Zone**.

By default, the Server default time zone is selected.

- b. Select the **Start date and time**.
- c. Select the **End date and time**.

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

- a. Select the Time Zone. By default the server time zone is selected.
- b. Select when to start. 1.Now 2.On a date and time - Select a date and time.
- c. Configuring the recurrence pattern to set up relative run Exclusion. Any recurrence pattern you set begins after the start time and End Time you specify. Recurrence pattern options are 1.For weekly user should be able to select one or more day of the week, associated with a start date end time. 2.For monthly user should be able to select a day in a month, associated with a start and end time. 3.For yearly user should be able to select a day in the year, associated with a start and end time.
- d. Select Stop after n occurrences - Relative Run Exclusion Rule stop after the specified number of runs have occurred (whether the Relative Run Exclusion Rule succeed or fail)
- e. Select Stop by a date and time - Relative Run Exclusion Rule are initiated as many times as defined until the specified end date and time is reached.



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

8. On the **Eligible schedules** tab, select the schedule for which you want to apply the exclusion rule by completing the following steps:
 - a. Search for the available schedules by entering a wildcard search in the **Filter** text box.

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

- c. Click .

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

- d. Click **Save**.

9. Click **Save**.

Deleting exclusion rules

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

To delete an exclusion rule, complete the following steps:

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



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

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

Enabling and disabling exclusion rules

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

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

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

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

The status of the rule changes to Enabled.

3. To disable a rule, select an enabled rule.
4. Click **Disable**.

The status of the rule changes to Disabled.

Importing exclusion rules

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

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

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

To import exclusion rules, complete the following steps:

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

Understanding the XML file to import exclusion rules

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

Tags in the XML file

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

Table 24. Tags in the XML file

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

Table 24. Tags in the XML file (continued)

Tag	Description
ruleEndTime	Time at which the exclusion rule ends. The format of the time must be HH:M-M:SS.
SchedulerID	IDs of the scheduler on which the exclusion rule must be applied. Multiple scheduler task IDs can be specified. The IDs of the scheduler tasks are available in the <code>USCH_TASK</code> table in the database.
ruleStatus	Status of the exclusion rule. The value can be <code>Enabled</code> or <code>Disabled</code> .

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

Example of the XML file to import exclusion rules

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

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

```
<rules>
  <rule>
    <ruleName>Rule1</ruleName><!-- specify rule name -->
    <ruleDescription>Rule for skipping 1/13 to 1/19.</ruleDescription><!-- specify rule description -->
    <ruleStartDate>1/13/2017</ruleStartDate><!-- specify exclusion start date. This should be of format
MM/DD/YYYY -->
    <ruleStartTime>8:00:00</ruleStartTime><!-- specify exclusion start time. This should be of format
HH:MM:SS-->
    <ruleEndDate>1/19/2017</ruleEndDate><!-- specify exclusion end date. This should be of format MM/DD/YYYY
-->
    <ruleEndTime>18:15:00</ruleEndTime><!-- specify exclusion end time. This should be of format HH:MM:SS -->
    <SchedulerIDs>
      <SchedulerID>10</SchedulerID> <!-- specify scheduler task IDs, on which this rule should get applied. This
needs to be obtained from database. -->
      <SchedulerID>15</SchedulerID>
    </SchedulerIDs>
    <ruleStatus>Enabled</ruleStatus> <!-- specify exclusion rule status. valid values Enabled/Disabled -->
  </rule>
</rules>
<rules>
  <rule>
```

```

<ruleName>Rule2</ruleName><!-- specify rule name -->
<ruleDescription>Rule for skipping 2/6 to 2/10</ruleDescription><!-- specify rule description -->
<ruleStartDate>2/6/2017</ruleStartDate><!-- specify exclusion start date. This should be of format
MM/DD/YYYY -->
<ruleStartTime>00:00:00</ruleStartTime><!-- specify exclusion start time. This should be of format
HH:MM:SS-->
<ruleEndDate>2/10/2017</ruleEndDate><!-- specify exclusion end date. This should be of format MM/DD/YYYY
-->
<ruleEndTime>23:59:59</ruleEndTime><!-- specify exclusion end time. This should be of format HH:MM:SS -->
<SchedulerIDs>
  <SchedulerID>45</SchedulerID> <!-- specify scheduler task Ids, on which this rule should get applied. This
needs to be obtained from database. -->
  <SchedulerID>88</SchedulerID>
</SchedulerIDs>
<ruleStatus>Disabled</ruleStatus> <!-- specify exclusion rule status. valid values Enabled/Disabled -->
</rule>
</rules>

```

What to consider when you use the scheduler with Unica Campaign

Some special configuration applies when you use the Unica Scheduler with Unica 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 Unica Campaign flowchart triggers. Triggers sent by the Schedule process or by the Unica Campaign trigger utility `unica_actrg` cannot cause schedules in the Unica Scheduler to run, and vice versa.

Difference between the Unica Campaign Schedule process and Unica Scheduler

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

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

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

Do not use the Unica Scheduler to schedule a flowchart that uses the Unica 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 Unica Scheduler, it functions as configured; conditions required by the Unica Scheduler and the Schedule process must be met before subsequent processes run.

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

The flowchart is designed to automatically retry a failed process box, a certain number of times with a specified delay between each retry, but only during scheduled runs. The flowchart will mark the status as failed if it is unable to retrieve a database connection within the retry interval.

The configuration parameters for the number of retries (`numberOfRetries`) and delay between retries (`delayBetweenRetries`) are customizable under the settings `Campaign|partitions|partition[n]|server|flowchartRun`, with a default value of zero. By default, the flowchart will not automatically retry any failed process boxes during scheduled runs.

It is important to note that this behaviour only applies to scheduled runs, so if the flowchart runs manually or triggered in some other way, the retry mechanism may not be activated. Additionally, if the retry interval is too long or the number of retries is too high, it could potentially cause performance issues or other problems, so these parameters should be carefully considered and adjusted as required.

Permissions for scheduling flowcharts

Scheduling Unica Campaign flowcharts using the Unica Scheduler requires the following permissions.

Table 25. 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 Unica 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

To schedule a flowchart using the default parameters, complete the following steps.

1. On the **Flowchart** tab in **View** mode, click the **Schedules** icon and select **Schedule This**. This opens the Override Flowchart Parameters window. All parameters are optional in this screen.
2. Click the **Schedule a Run** button located at the lower pane. This opens a window which allows you to schedule a flowchart using default parameters.
3. Complete the fields in the **Schedule flowchart** box. If you choose to run more than once, click **Set up Recurrences** to set up a recurrence pattern.
4. Click **Run** with this schedule.

About overriding the default parameters for Unica Campaign flowchart run schedules

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

When you schedule a Unica 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.

Unica 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 Unica Campaign flowcharts

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

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

Creating a flowchart schedule

Follow this procedure to schedule a flowchart.

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

The Override flowchart parameters for dialog box opens.

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

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

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

If you do not want to override default flowchart parameters, go on to the next step.

3. Click **Schedule a run** to open the Create a schedule dialog box.

You can define when the schedule runs and optionally set up recurrences, triggers, and throttling.

4. Click **Run with this schedule**.



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

The Override Flowchart Parameters page

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

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

Table 26. Fields on the Override Flowchart Parameters page

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

Schedule notifications

You can set up notifications for any schedule, to alert you to the status of scheduled runs. In addition, users with Administrator permissions in Unica 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 Unica 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.

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

Deleting or modifying notifications for schedules you create

You can delete or modify any notifications you created.

1. Select **Settings > My job notifications** to open the My job 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 Unica Platform to perform this procedure.

1. On the **Settings > Configuration** page, go to the **Unica Platform | Scheduler | Schedule registrations** category.
2. For each object type for which you want to enable group-based notifications, set the value of the **Name of group to receive job notifications** property to the name of the group you want to receive notifications for this object type.

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

You might want to set up a group for each object type for which you want to enable group-based notifications.

3. On the User groups page, assign users to the group or groups that you specified in the previous step, as required.

The My job notifications page

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

Table 27. Fields on the My job notifications page

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

Schedule management

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

These are the tabs on the Scheduled tasks page.

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

Schedules and partitions

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

Unknown status

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

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

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

The schedule list filter

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

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

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

Disabling and enabling multiple schedules (with FixPack 10.0.0.1 only)

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

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

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

The Schedule management pages

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

The Schedules tab

Table 28. Fields and links on the Schedules tab






Field or link	Description
 Disable	Disable one or more selected schedules. Available only if you have applied Unica Platform FixPack 10.0.0.1.
 Enable	Enable one or more selected schedules. Available only if you have applied Unica Platform FixPack 10.0.0.1.
Create a schedule	Click to open a wizard where you can set up a schedule.
Edit the schedule list filter	Click to create an advanced filter for the list.
Delete	<p>Delete one or more selected schedules. You can select schedules by clicking in the column at the left of the schedule. To select all schedules, click at the top of the left side column.</p> <p> Note: Mailing type of schedules cannot be deleted using this button. They can only be deleted using the cancel schedule option on Mailing page.</p>
Refresh	Click to refresh the list.
Filter	Click to create a simple filter for the list.
Schedule name	The schedule of which the run is an instance.
Schedule state	Whether the schedule is enabled or disabled.
Scheduled item	The name of the object to be run.
Item type	The type of object to be run.
Created by	The user name of the account that created the schedule.
Start trigger	If the schedule depends on a trigger, the trigger that causes the schedule to run.
Start	Date and time when the first run of this task is scheduled.
Recurrence pattern	A description of the recurrence pattern.
End	Date and time when the last run of this task is scheduled.

Table 28. Fields and links on the Schedules tab (continued)

Field or link	Description
	 Note: Applies to recurring scheduled tasks only.
Previous 1 and next 2 runs	Date and time of the previous run and next two scheduled runs.  Note: Applies to recurring scheduled tasks only. The information about previous one and next two scheduled runs is shown as per the scheduler definition. It is not currently validated against the exclusion rules.
Dependencies	If the scheduled object depends on other objects, they are listed here.
On success trigger	The string that is sent if the product reports that a run of this schedule has completed successfully. This field is blank if no on success trigger is specified.
On failure trigger	The string that is sent if the product reports that a run of this schedule has failed. This field is blank if no on failure trigger is specified.

The Runs tab

Table 29. Fields and links on the Runs tab

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

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

Field or link	Description
Execution state	<p>State of the run as defined in the scheduler, as follows.</p> <ul style="list-style-type: none"> • Scheduled - The run has not begun. • Queued - The scheduler has started the run, but the Unica product has not begun executing the scheduled run due to throttling constraints. • Completed - The run has completed and has returned a status of Failed or Succeeded. • Cancelled - A user has cancelled a run by clicking Mark as Cancelled on the Scheduled Runs page. If the run was queued when the user marked it as cancelled, it does not execute. If the run was executing, this action does not stop the run, but it is marked as cancelled, and any triggers configured for this run are not sent. Also, runs that depend on the cancelled run do not execute. • Unknown - Indicates that Unica Platform can not determine whether the job is still running, completed or failed.
Run status	<p>Status of the object's run as defined by the product executing the run. If the run sends a status of Cancelled, and the run is later started again and sends any other status to the scheduler, the status is updated in this field.</p>
Details	<p>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.</p>

Edit the schedule list filter - Schedules

Table 30. Edit the schedule list filter on the Schedules tab

Column	Description
Filter by search tags / keywords	<p>Select this check box if you want to include search tags or keywords in your filter. The string you enter here is matched with strings entered in the Search tags / keywords fields when schedules are created.</p>
Search tags / keywords	<p>Enter the search tags or keywords you want to use in your filter.</p>
Filter on other criteria	<p>Select this check box if you want to include additional criteria in your filter.</p>
Run metadata	<p>Select one of the following options to include in your rule.</p> <p>The options are:</p> <ul style="list-style-type: none"> • Schedule name • Schedule state

Table 30. Edit the schedule list filter on the Schedules tab (continued)

Column	Description
	<ul style="list-style-type: none"> • Item type • Created by • Scheduled item
Condition	<p>Select one of the following options to determine how your rule is evaluated.</p> <ul style="list-style-type: none"> • Matches • Starts with • Ends with • Contains
Value	<p>Enter or select the value you want to apply to the rule. The options vary depending on the metadata you select for the rule.</p> <ul style="list-style-type: none"> • Schedule name Enter any characters. • Schedule state Value options are Enabled and Disabled. • Item type Value options are the various schedule types. • Created by Enter any characters. Your value is compared with user login names. • Scheduled item Enter any characters. The string you enter here is compared with the text in the Scheduled item column.
And / Or	Select one of these operators for each rule you create.

Edit the schedule list filter - Runs**Table 31. Edit the schedule list filter on the Runs tab**

Column	Description
Filter based on time	Select this check box if you want to show runs that occurred within a specified time interval.

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

Column	Description
Time zone	If you select an option other than the server default, the search uses the selected time zone to calculate which schedules fall within the date range you specify.
List runs for the last n instances	For recurring runs, specify how many previous runs to show in the list.
List runs from	Specify a time interval for the runs shown in the list.
Filter on other criteria	Select this check box if you want to include additional criteria in your filter.
Run metadata	<p>Select one of the following options to include in your filter.</p> <p>The options are:</p> <ul style="list-style-type: none"> • Schedule name • Execution state • Run status • Scheduled item
Condition	<p>Select one of the following options to determine how your criteria are evaluated.</p> <ul style="list-style-type: none"> • Matches • Starts with • Ends with • Contains
Value	<p>Enter or select the value you want to apply to the filter. The options vary depending on the metadata you select for the rule.</p> <ul style="list-style-type: none"> • Schedule name <p>Enter any characters.</p> <ul style="list-style-type: none"> • Execution state <p>Value options are:</p> <ul style="list-style-type: none"> ◦ Queued ◦ Running ◦ Completed ◦ Unknown ◦ Cancelled <ul style="list-style-type: none"> • Run status

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

Column	Description
	<p>Value options are Succeeded, Running, Cancelled,Failed, and Unknown.</p> <ul style="list-style-type: none"> • Scheduled item <p>Enter any characters. The string you enter here is compared with the text in the Scheduled item column.</p>
And / Or	Select one of these operators for each rule you create.

SAML 2.0 based federated authentication

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

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

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

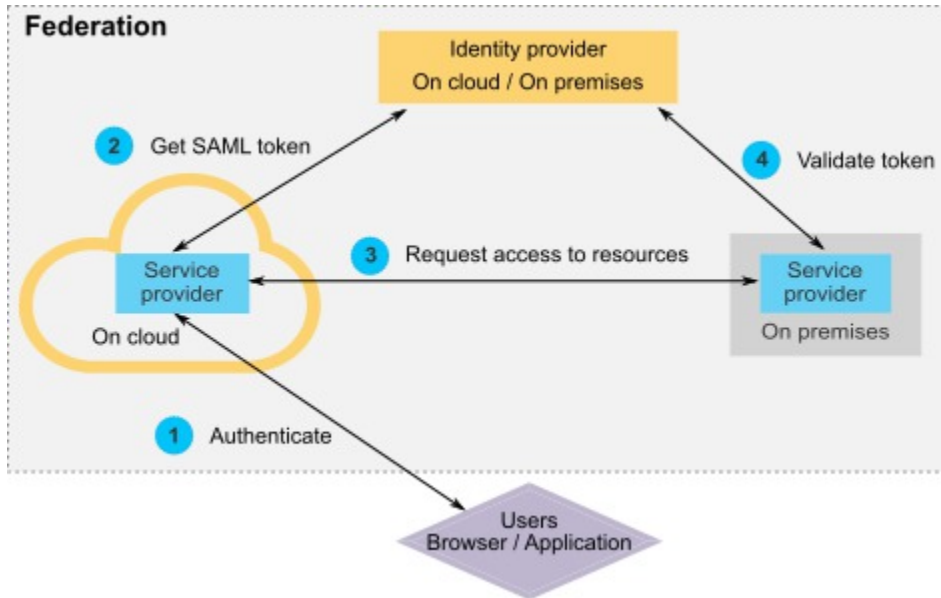
A SAML 2.0 federation supports a variety of authentication mechanisms for single sign-on. For example, a user can be authenticated in an SP using that application's authentication mechanism (for example, in-house, OAuth, OpenId, SAML, Kerberos), and then the user can access other SPs using federated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

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

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

Diagram

The following diagram illustrates the federation.



Components of the HCL implementation

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

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

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

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

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

Supported use cases

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

Generating a new SAML token

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

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

Validating an existing SAML token

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

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

Deleting an existing SAML token

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

Limitations

The current implementation does not support the following use cases.

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

Federated authentication and partitions

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

How to implement federated authentication

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

Creating the data repository

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

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

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

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

Table 32. Fields in the TP_MASTER table

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

Table 33. Fields in the TP_MAPPING table

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

Table 33. Fields in the TP_MAPPING table (continued)

Field	Description
LAST_REQUEST	Timestamp of the last successful request.

Configuring the IdP data source in the web application server

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

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

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

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

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

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

Setting up the classpaths for the IBM® IdP client façade

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

1. Obtain the required JAR files as described below, and place these JAR files on your IdP server and the servers that host your SPs.
 - Locate the `unica.war` file in the Unica Platform installation directory. Extract the `unica.war` file, navigate to the `WEB-INF\lib` directory and copy the following JARs.
 - `bcprov-jdk15on-1.68.jar`
 - `esapi.jar`

- `jersey-core-1.17.jar`
- `jersey-server-1.17.jar`
- `jersey-servlet-1.17.jar`
- `joda-time.jar`
- `opensaml.jar`
- `openws.jar`
- `xmlsec.jar`
- `xmltooling.jar`
- `asm-3.1.jar`

Download from <http://mvnrepository.com/artifact/asm/asm/3.1>.

- `jcl-over-slf4j-1.7.5.jar`

Download from <http://mvnrepository.com/artifact/org.slf4j/jcl-over-slf4j/1.7.5>.

- `slf4j-api-1.7.5.jar`

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

2. Add the JAR files you obtained in the previous step in the classpath of your IdP server and in the classpath of each of your SPs.

3. For each SP that you want to include in the federation, also add this Client façade JAR file the classpath: :

`idp-client.jar`

This JAR file is provided with your Unica Platform installation.

Deploying the IdP server

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

Configuring the IdP server

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

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

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

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

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



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

Reference: IdPServerConfig.properties file

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

`com.ibm.ocm.idp.server.keystore.path`

The absolute path of the keystore file on the web application server host machine.

Default value: path/idp.jks

`com.ibm.ocm.idp.server.keystore.passkey`

Passkey of the keystore.

Default value: idp001

`com.ibm.ocm.idp.server.keystore.alias`

Alias of the keystore.

Default value: idp

`com.ibm.ocm.idp.server.certificate.issuer`

Certificate issuer's URL.

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

`com.ibm.ocm.idp.server.token.validity`

Token validity period in seconds.

Default value: 3600

`com.ibm.ocm.idp.server.enable`

Logger for IdP server.

Default value: True

`com.ibm.ocm.idp.server.dao.class`

Database specific data access object implementation.

Supported DAOs are:

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

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

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

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

`com.ibm.ocm.idp.server.datasource.name`

JNDI data source name defined in the application server.

Default value: `idp_datasource`

`com.ibm.ocm.idp.server.query.token.create`

Query to create token.

Default value:

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

`com.ibm.ocm.idp.server.query.token.get`

Query to get token.

Default value:

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

`com.ibm.ocm.idp.server.query.mapping.validate`

Query to validate a user mapping.

Default value:

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

`com.ibm.ocm.idp.server.query.token.delete`

Query to delete token.

Default value:

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

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

Query to get client details.

Default value:

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

Obtaining keystores and importing them into the IdP server

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

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

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

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

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

- An SP administrator generates a keystore and exports it.

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

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

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

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

Setting configuration properties on the Configuration page

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

Set configuration properties under the following categories.

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

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

Onboarding Service Providers and users

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

Here is example SQL for onboarding an SP.

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

After the trusted parties are registered with the IdP server, the IdP server administrator can map users to participate in federated single sign-on.

The user mapping must be strictly one-to-one between two SPs. For example, User1 from SP_A must be mapped ONLY to any one user in SP_B. However, User1 from SP_A can be mapped with another user in SP_C in the same federation.

Here is an example query for adding users in the `TP_MAPPING` table.

```
INSERT INTO TP_MAPPING
(TP_CLIENT_ID, TP_FOR_USER_ID, TP_SP_ID, TP_MAPPED_USER_ID, SAML_TOKEN)
VALUES
('SP1_Id', 'SP1_user_Id', 'SP2_Id', 'SP2_user_id', 'dummy1')
```



Note: The entries for `TP_SP_ID` and `TP_FOR_USER_ID` must be a minimum of 4 characters and up to 24 characters long, and contain only alphanumeric, hyphen and underscore characters: `[a-zA-Z0-9_-]`. Insert unique dummy entries for the `SAML_TOKEN` column, as this column does not allow nulls and duplicates.

Using the IdP client façade to generate tokens and pass them to Service Providers

When a user is authenticated and wants to access the services of another SP, call the following code on the SP side.

The code generates the federated token.

```
// One time properties to initialize the IdP client.
Properties properties = new Properties();
properties.put(IdPClient.IDP_SERVER_URL, "URL");
properties.put(IdPClient.IDP_CLIENT_CERTIFICATE_ISSUER, "URL");
properties.put(IdPClient.IDP_CLIENT_KEYSTORE_PATH, "JKS file path");
properties.put(IdPClient.IDP_CLIENT_KEYSTORE_PASSKEY, "JKS passkey");
properties.put(IdPClient.IDP_CLIENT_KEYSTORE_ALIAS, "Certificate alias");
// Get the IdP client factory singleton instance
//with the specified parameters.
IdPClientFactory clientFactory = IdPClientFactory.getInstance(properties);
// Get the partition specific client facade to do the assertion.
IdPClientFacade clientFacade = clientFactory.getIdPClientFacade(partition);
// Establish SSO Login with the IdP server
IdPClientToken token = clientFacade.doIdPLogin(clientId, forUserId, spId);
```

After the token is obtained, it can be passed to target SPs to access their resources based on the mapped user's roles and permissions.

```
// Security token is validated at Service Provider side.
IdPClientAssertion assertion = spFacade.assertIdPToken(clientId, forUserId, spId,
token.getTokenId());
// Retrieve the principal from the assertion, if there is no exception.
String principal = assertion.getMappedUser();
```

The client facade is multi-tenant aware and can be used to configure each partition separately. To use this feature, append the client ID to each property name. For example:

```
properties.put(IdPClient.IDP_CLIENT_KEYSTORE_PATH +
".partition1", "JKS file path");
properties.put(IdPClient.IDP_CLIENT_KEYSTORE_PASSKEY +
".partition1", "JKS passkey");
```

```
properties.put(IdPClient.IDP_CLIENT_KEYSTORE_ALIAS +
".partition1", "Certificate alias");
```

Reference: RESTful services

Use this information to troubleshoot issues when you use the client facade, or to develop your own SAML 2.0 implementation with the IdP server provided by IBM.

The REST APIs are implemented using an XML data payload. The SAML assertion is directly passed to the POST methods with digital signatures.

Only the POST method is supported for all verbs to ensure unified method access and to enforce security assertions, based on the XML payload. Other methods, such as GET, PUT, and DELETE, return an error message. The following table represents the verbs that implement the supported use cases.

Table 34. Supported verbs

Resource	Post
<idp>/saml/token/clientId/forUserId/spId/create	Generate new SAML token.
<idp>/saml/token/clientId/forUserId/spId/validate	Validate existing SAML token.
<idp>/saml/token/clientId/forUserId/spId/delete	Delete existing SAML token.

Related concepts

This section provides general information about the technologies used in the ExperienceOne implementation of SAML 2.0 based federated single sign-on.

Security Assertion Markup Language 2.0 (SAML 2.0)

SAML 2.0 is a version of the SAML standard for exchanging authentication and authorization data between security domains. SAML 2.0 is an XML-based protocol that uses security tokens containing assertions to pass information about a principal (usually an end user) between a SAML authority, that is, an identity provider, and a SAML consumer, that is, an SP. SAML 2.0 enables web-based authentication and authorization scenarios including cross-domain single sign-on (SSO), which helps reduce the administrative overhead of distributing multiple authentication tokens to the user. For more information, see http://en.wikipedia.org/wiki/SAML_2.0.

Identity Provider (IdP)

Also known as Identity Assertion Provider, the IdP issues identification information for all SPs that interact or provide services within the system. This is achieved via an authentication module that verifies a security token as an alternative to explicitly authenticating a user within a security realm. In perimeter authentication, a user needs to be authenticated only once (single sign-on) and pass along a security token which is processed by an Identity Assertion Provider for each system it needs to access. For more information, see http://en.wikipedia.org/wiki/Identity_provider.

Public-key cryptography

Also known as asymmetric cryptography, a cryptographic algorithm that requires two separate keys, one of which is secret (or private) and one of which is public. Although different, the two parts of this key pair are mathematically linked. The public key is used to encrypt plaintext or to verify a digital signature; whereas the private key is used to decrypt ciphertext or to create a digital signature. For more information, see http://en.wikipedia.org/wiki/Public-key_cryptography.

SAML 2.0 single sign-on

Unica Platform supports SAML 2.0 based single sign-on.

In this mode, Unica users can be authenticated against any external or corporate identity provider that follows the standard SAML 2.0 protocol. Identity providers generate the SAML assertion, which is then used by Unica Platform to allow users to log in. Therefore, a fully functional SAML 2.0 IdP server is a prerequisite for this integration.

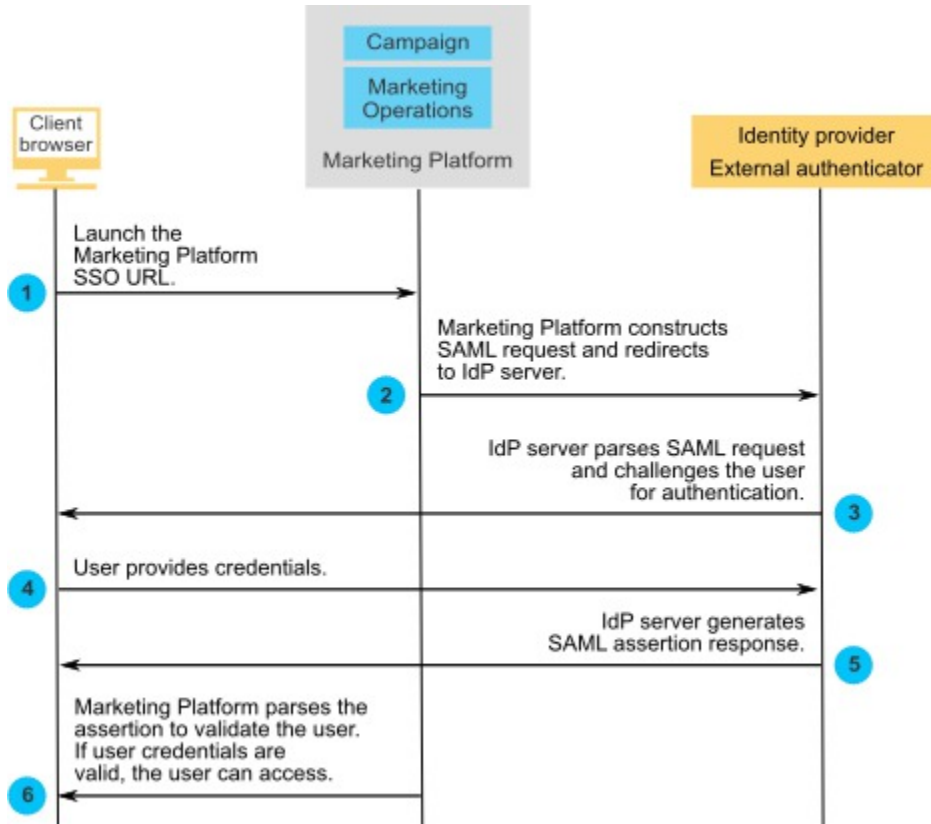
After you set up the required configuration properties and a metadata file, users who attempt to log in through the Unica Platform login page are authenticated through your organization's SAML 2.0 Identity Provider (IdP) server.

A configuration property, **Add authenticated users to Platform**, enables automatic creation of a Unica Platform account for any authenticated user who does not have a Unica Platform account. These users are automatically added to a default user group, **ExternalUsersGroup**, which has only the **PlatformUser** role initially. Alternatively, you can specify a custom group to which users are added.

If the **Add authenticated users to Platform** property is not enabled, users must have a Unica Platform account to log in.

A Unica Platform administrator can manage group memberships and roles to configure access to Unica products for the automatically created users.

The following diagram illustrates the SAML 2.0 based single sign-on mode in Unica.



Behavior when SAML 2.0 single sign-on is implemented

The implementation of SAML 2.0 single sign-on in Unica has the following behavior.

Logout

When logged-in users log out, they are redirected to the standard Unica logout page. A message instructs users to close the browser window to complete the log out process.

Session timeout

If logged-in users are idle for more than 30 minutes they are redirected to the standard Unica logout page. A message instructs users to close the browser window to complete the log out process.

This time out period can be configured in your application server.

Incorrect SAML configuration

If an error occurs due to incorrect SAML configuration when users attempt to log in, the users are redirected to an error page with the message "Login Failed: Bad Credential. Please close the browser window and try again."

User not provisioned in Platform but is a valid IDP user

When the **Add authenticated users to Platform** property is disabled, users who log in with credentials that are valid for the IdP server but who do not have a Unica Platform account are redirected to an error page with the message "Login Failed: Bad Credential. Please close the browser window and try again."

User exists in both IDP and Unica Platform but their Unica Platform password has expired or been reset

When user's password is expired or reset in Unica Platform, the user is redirected to an error page with the message "Login Failed: Bad Credential. Please close the browser window and try again."

Configuration process roadmap: SAML 2.0 single sign-on

Use this configuration process roadmap to scan the tasks required to implement SAML 2.0 single sign-on in Unica. Where applicable, the Task column provides links to the topics that describe the tasks in detail.

Table 35. Configuration process roadmap: SAML 2.0 single sign-on

Task	Information
Add Unica Platform as a service provider in your IdP server. See your IdP server documentation for details.	<ul style="list-style-type: none"> • Make a note of the application ID assigned to Unica Platform in your IdP server, as you will use it as the value of a configuration property. • Make a note of password of the key store file, as you will use it as the value of the password in the data source you create.
Setting up the metadata file (on page 112)	Set up the XML metadata file generated by your SAML 2.0 IdP server.
Setting SAML 2.0 configuration properties (on page 113)	Set required configuration properties on the Settings > Configuration page .
Setting up a data source for SAML single sign-on (on page 113)	Set up the a data source to hold the password of the key-store file.

Setting up the metadata file

Your IdP server generates a metadata file that contains configuration and integration details for SAML 2.0 single sign-on.

Copy the IdP server metadata file and place it on the server where Unica Platform is installed. Make a note of the following information, which you need when you set configuration property values.

- The location of the metadata file on the Unica Platform server.
- The value of *entityID* in the XML declaration at the top of the metadata file.

Setting SAML 2.0 configuration properties

To configure SAML 2.0 single sign-on, set properties on the **Configuration > Settings** page.

Set the following properties.

- Set the value of the **Login method** property to **SAML 2.0**.

This property is located under the **Unica Platform | Security** node.

Stop and restart the Unica Platform web application so that this change takes effect.

- Set properties located under the **Unica Platform | Security | Login method details | SAML 2.0** node as required.

See the context help for these properties for details.

Setting up a data source for SAML single sign-on

Save the password of the key store file in a data source in Unica Platform.

1. Log in to Unica as an Admin user and navigate to the Settings > Users page.
2. Select or create a user and configure a data source for this user as follows.
 - **Data Source** - Enter the value set for the **Key store credential data source** property under **Unica Marketing Platform | Security | Login method details | SAML 2.0** on the **Settings > Configuration** page.
 - **Data Source Login** - Enter the value set for the **Key store credential holder** property under **Unica Platform | Security | Login method details | SAML 2.0** on the **Settings > Configuration** page.
 - **Data Source Password** - Enter the password of the key store file used for Unica Platform in your IdP server.

If you have multiple partitions, you must perform this task in every partition where you have users who should have single sign-on. Alternatively, you can use the platform_admin user account for this step. Because this user is a member of all partitions, the data source is available in all partitions

Sample SAML 2.0 IdP assertion

This section provides examples of the SAML 2.0 request and response.

Example of the SAML 2.0 request generated by Unica Platform

Unica Platform generates the SAML 2.0 request shown in this section, and encodes it using OpenSAML Base64 APIs. The request is compatible with any other standard Base64 decoder. The encoded request is posted to the IdP server.



```

<saml2p:AuthnRequest
  xmlns:saml2p="urn:oasis:names:tc:SAML:2.0:protocol"
  AssertionConsumerServiceURL="http://example.com"
  Destination="http://example.com"
  ForceAuthn="false"
  ID="_0ff13d123291170422ff5e945e9a209e25f3404916451a4aaf"
  IsPassive="false"
  IssueInstant="2015-09-02T14:10:24.376Z"
  ProtocolBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
  Version="2.0">
  <saml2:Issuer xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">
    IdP_ID
  </saml2:Issuer>
  <saml2p:NameIDPolicy
    AllowCreate="true"
    Format="urn:oasis:names:tc:SAML:2.0:nameid-format:transient"
    SPNameQualifier="SERVICE_PROVIDER_ID"/>
  <saml2p:RequestedAuthnContext
    Comparison="exact">
    <saml2:AuthnContextClassRef
      xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">
      urn:oasis:names:tc:SAML:2.0:ac:classes:
      PasswordProtectedTransport
    </saml2:AuthnContextClassRef>
  </saml2p:RequestedAuthnContext>
</saml2p:AuthnRequest>

```

Example of the SAML 2.0 response generated by the IdP server

```

<samlp:Response
  xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
  Destination="http://serviceprovider.com/location"
  ID="id-wmpfMj-fMh0ihGYJ73rXPTEq7o8-"
  InResponseTo="s2e211c5bfc0200fc48819f381f17d56ca0b5c780f"
  IssueInstant="2015-09-02T14:10:24.376Z"
  Version="2.0">
  <saml:Issuer xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
    Format="urn:oasis:names:tc:SAML:2.0:nameid-format:entity">
    Identity Provider
  </saml:Issuer>
  <samlp:Status>

```

```

    <samlp:StatusCode Value="urn:oasis:names:tc:
      SAML:2.0:status:Success" />
  </samlp:Status>
  <saml:Assertion xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
    ID="idzQ07U5TzPLLL4dlqTqRt9VI01Yg-"
    IssueInstant="2015-09-02T14:10:24.376Z"
    Version="2.0">
    <saml:Issuer Format="urn:oasis:names:tc:SAML:2.0:
      nameid-format:entity">
      Identity Provider
    </saml:Issuer>
    <dsig:Signature xmlns:dsig="http://www.w3.org/2000/09/xmldsig#">
      <dsig:SignedInfo>
        <dsig:CanonicalizationMethod
          Algorithm="http://www.w3.org/2001/10/xmlexc-c14n#" />
        <dsig:SignatureMethod
          Algorithm="http://www.w3.org/2000/09/xmldsig#
            rsa-sha1" />
        <dsig:Reference URI=
          "#id-zQ07U5TzPLLL4dlqTqRt9VI01Yg-" />
        <dsig:Transforms>
          <dsig:Transform Algorithm=
            "http://www.w3.org/2000/09/xmldsig#
              enveloped-signature" />
          <dsig:Transform Algorithm=
            "http://www.w3.org/2001/10/xml-exc-c14n#" />
        </dsig:Transforms>
        <dsig:DigestMethod Algorithm=
          "http://www.w3.org/2000/09/xmldsig#sha1" />
        <dsig:DigestValue>
          XXX=
        </dsig:DigestValue>
        </dsig:Reference>
      </dsig:SignedInfo>
      <dsig:SignatureValue>xxx</dsig:SignatureValue>
    </dsig:Signature>
    <saml:Subject>
      <saml:NameID Format=
        "urn:oasis:names:tc:SAML:2.0:nameid-format:
          transient"
        NameQualifier="Test Identity Provider"
        SPNameQualifier="TEST">

```

```

        id-N2EIOvbwaVf1UP-cKTzgv8dGYLg-
    </saml:NameID>
    <saml:SubjectConfirmation
        Method="urn:oasis:names:tc:SAML:2.0:cm:bearer">
        <saml:SubjectConfirmationData
            InResponseTo=
                "s2e211c5bfc0200fc48819f381f17d56ca0b5c780f"
            NotOnOrAfter="2015-09-02T14:10:24.376Z"
            Recipient="http://serviceprovider.com/location" />
        </saml:SubjectConfirmation>
    </saml:Subject>
    <saml:Conditions
        NotBefore="2015-09-02T14:10:24.376Z"
        NotOnOrAfter="2015-09-02T14:10:49.376Z">
        <saml:AudienceRestriction>
            <saml:Audience>TEST</saml:Audience>
        </saml:AudienceRestriction>
    </saml:Conditions>
    <saml:AuthnStatement
        AuthnInstant="2015-09-02T14:10:24.376Z"
        SessionIndex="id-1FTYalkjaVTWwHrFRkIREvHfAxx-"
        SessionNotOnOrAfter="2015-09-02T14:10:38.376Z">
        <saml:AuthnContext>
            <saml:AuthnContextClassRef>
                urn:oasis:names:tc:SAML:2.0:ac:classes:
                PasswordProtectedTransport
            </saml:AuthnContextClassRef>
        </saml:AuthnContext>
    </saml:AuthnStatement>
    <saml:AttributeStatement xmlns:x500=
        "urn:oasis:names:tc:SAML:2.0:profiles:
        attribute:X500"
        xmlns:xs="http://www.w3.org/2001/XMLSchema"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <saml:Attribute
            Name="UserIdentifier"
            NameFormat="urn:oasis:names:tc:SAML:2.0:
            attrnameformat:basic">
            <saml:AttributeValue xsi:type="xs:string">
                user@example.com
            </saml:AttributeValue>
        </saml:Attribute>
    </saml:Attribute>

```

```

    </saml:AttributeStatement>
  </saml:Assertion>
</samlp:Response>

```

Sample IdP metadata

This section provides an example of the metadata file produced by the IdP server

Example of the metadata file generated by the IdP server

The IdP server generates a metadata file that contains configuration and integration details for SAML 2.0 single sign-on. This file is used by Unica Platform. Place a copy of the file generated by your IdP server on the Unica Platform server.

```

<?xml version="1.0" encoding="UTF-8"?>
  <md:EntityDescriptor xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
    entityID="ENTITY_ID">
    <md:IDPSSODescriptor
      WantAuthnRequestsSigned="false"
      protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:
        protocol">
      <md:KeyDescriptor use="signing">
        <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
          <X509Data>
            <X509Certificate>
              __certificate__
            </X509Certificate>
          </X509Data>
        </KeyInfo>
      </md:KeyDescriptor>
      <md:KeyDescriptor use="encryption">
        <KeyInfo
          xmlns="http://www.w3.org/2000/09/xmldsig#">
          <X509Data>
            <X509Certificate>
              __certificate__
            </X509Certificate>
          </X509Data>
        </KeyInfo>
        <md:EncryptionMethod
          Algorithm=
            "http://www.w3.org/2001/04/xmlenc#rsa-1_5" />

```

```

</md:KeyDescriptor>
<md:ArtifactResolutionService
  Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
  Location="__location__" index="0"
  isDefault="true"/>
<md:ArtifactResolutionService
  Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
  Location="__location__" index="1"/>
<md:SingleLogoutService
  Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
  Location="__location__"/>
<md:SingleLogoutService Binding=
  "urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
  Location="__location__"/>
<md:ManageNameIDService
  Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
  Location="__location__"/>
<md:ManageNameIDService
  Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
  Location="__location__"/>
<md:NameIDFormat>urn:oasis:names:tc:SAML:2.0:
  nameid-format:persistent
</md:NameIDFormat>
<md:NameIDFormat>urn:oasis:names:tc:SAML:2.0:
  nameid-format:transient
</md:NameIDFormat>
<md:NameIDFormat>
  urn:oasis:names:tc:SAML:1.1:
  nameid-format:emailAddress
</md:NameIDFormat>
<md:NameIDFormat>urn:oasis:names:tc:SAML:2.0:
  nameid-format:encrypted</md:NameIDFormat>
<md:SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:
  bindings:HTTP-POST" Location="__location__"/>
</md:IDPSSODescriptor>
<md:Organization>
  <md:OrganizationName xml:lang="en">
    organization_name
  </md:OrganizationName>
  <md:OrganizationDisplayName xml:lang="en">
    organization_display_name
  </md:OrganizationDisplayName>

```

```

        <md:OrganizationURL xml:lang="en" />
    </md:Organization>
    <md:ContactPerson contactType="technical">
        <md:Company>
            company
        </md:Company>
        <md:GivenName/>
        <md:SurName/>
        <md:EmailAddress/>
        <md:TelephoneNumber/>
    </md:ContactPerson>
</md:EntityDescriptor>

```

Single sign-on between Unica and IBM Digital Analytics

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

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

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

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

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

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

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

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

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

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

Permissions in Digital Analytics for single sign-on users

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

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

- By default, users have the permissions granted to the Digital Analytics group the administrator has configured for all automatically created users.

Administrators can modify the permissions associated with this group.

- In addition, the administrator can override automatic account creation for users who already have a Digital Analytics account. If the override is in place for a user, that user has the permissions he or she would have when he or she logs in to Digital Analytics directly.

Server clock coordination

The clock on the server on which Unica Platform is deployed must match the time on the Digital Analytics server clock. For single sign-on, the Digital Analytics server allows for up to 15 minutes of difference (900 seconds) between server clock times.

As a best practice, you should synchronize server clocks. To ensure synchronization, you should use the Network Time Protocol (NTP).

If you cannot synchronize your server clock, and there might be at least 15 minutes of difference between the clocks, you can set the **Clock skew adjustment (seconds)** configuration property under the Coremetrics® category in Unica Platform to a number that reflects the difference between the clocks.

Setting up single sign-on between Unica and Digital Analytics using automatic user account creation

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

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

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

2. Log in to Digital Analytics as an Admin user with access to the Client ID you selected in the previous step, click the Admin link, and navigate to the Global User Authentication page.
 - In the **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 Unica 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 Unica 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 Unica Platform, navigate to the **Settings > Configuration** page and set configuration properties as follows.

Table 36. Configuration properties for enabling single sign-on with Digital Analytics

Property	Value
Digital Analytics Enable IBM Digital Analytics	True
Digital Analytics Integration partitions partition[n] Platform user for IBM Digital Analytics account	Enter the login name for the Unica 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 Unica Campaign partition.

7. For any user for whom you want to override automatic account creation, do the following.

- In Unica Platform, navigate to the **Settings > Users** page.
- Enter the user's Digital Analytics login name in the **Digital Analytics Username** field on the user's detail page.

This works only for users who already have an Digital Analytics account.



Note: If an account does not exist in Digital Analytics with this login name, an account will be created for this user with the name you enter here, rather than with the user's Unica Platform login name.

8. Configure you web application server for single sign-on with Digital Analytics.

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

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

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

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

2. Log in to Digital Analytics as an Admin user with access to the Client ID you selected in the previous step, click the Admin link, and navigate to the Global User Authentication page.
 - In the **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 Unica 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 Unica 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 Unica Platform, navigate to the **Settings > Configuration** page and set configuration properties as follows.

Table 37. Configuration properties for enabling single sign-on with Digital Analytics

Property	Value
Digital Analytics Enable IBM Digital Analytics	True
Digital Analytics Integration partitions partition[n] Platform user for IBM Digital Analytics account	Enter the login name for Unica 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 Unica Campaign partition.

7. In Unica Platform, navigate to the **Settings > Users** page.
8. For each user for whom you want to enable single sign-on, enter that user's Digital Analytics login name in the **IBM Digital Analytics user name** field on the user's Edit properties page.



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

9. Configure your web application server for single sign-on with Digital Analytics.

Configuring WebLogic for single sign-on between Digital Analytics and Unica

Perform the procedure below in the WebLogic domain where Unica Platform is deployed to ensure that users can view Digital Analytics reports in dashboards without having to log in.

1. Open the `setDomainEnv` script, located in the `bin` directory under your WebLogic domain directory.
2. Add `-Dweblogic.security.SSL.ignoreHostnameVerification=true` to `JAVA_OPTIONS`.

Configuring WebSphere® for single sign-on between Digital Analytics and Unica

Perform the procedure below in WebSphere® cell and node where Unica Platform is deployed to ensure that users can view Digital Analytics reports in dashboards without having to log in.

1. Log in to the WebSphere® administrative console.
2. Expand **Security** and click **SSL certificate and key management**.
3. Under **Configuration settings**, click **Manage endpoint security configurations**.

4. Navigate to the outbound configuration for the cell and node where the Unica Platform is deployed.
5. Under **Related Items**, click **Key stores and certificates** and click the **NodeDefaultTrustStore** key store.
6. Under **Additional Properties**, click **Signer certificates** and **Retrieve From Port**.

Complete fields as follows.

- **Host name:** `welcome.coremetrics.com`
- **Port:** `443`
- **Alias:** `coremetrics_cert`

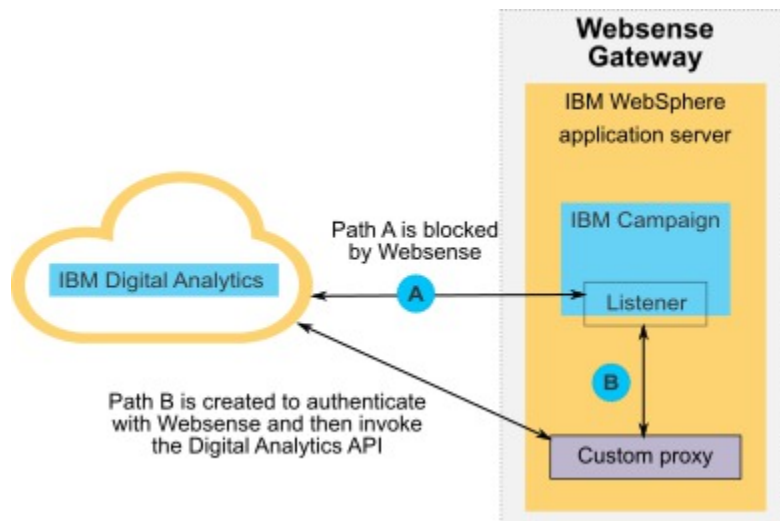
Digital Analytics integration with Websense using a custom proxy

Unica Platform provides a custom proxy to enable integration between Unica Campaign hosted on premises and Digital Analytics in the cloud when Websense is a required component of the environment.

The custom proxy is supported with the WebSphere application server only.

After the custom proxy is installed, you can configure single sign-on and integration between Digital Analytics and Unica Campaign.

The custom proxy is a Java servlet implementation that acts as a forward proxy. It is injected between the Unica Campaign listener and Digital Analytics. The custom proxy acts as an end point for the Unica Campaign listener to invoke Digital Analytics APIs. Internally, the custom proxy authenticates itself with the Websense content gateway and then securely invokes the APIs outside the network.



Deploying the custom proxy on WebSphere

Perform this procedure to install the custom proxy. This custom proxy is supported with the WebSphere application server only.

Note that you can deploy ProxyServer application in the same WebSphere Profile where you deployed Unica Campaign, or you can use different WebSphere profile.

1. Copy the `ProxyServer.war` file to a location that can be accessed from the WebSphere server.

You can find the `ProxyServer.war` file in the `tools\lib` directory under your Unica Platform installation.

2. Deploy the `ProxyServer.war` file, following these guidelines.
 - Select the **Detailed - Show all installation options and parameters** path for installation.
 - You can provide any application name.
 - You do not have to select **Precompile JavaServer Pages files**.
 - On the Initialize parameters for servlets page, complete the fields as shown below.
 - **proxy_host** - Host URL or IP address of the Websense server
 - **proxy_port** - Port number of the Websense server
 - **proxy_username** - User name for Websense authentication
 - **Proxy_password** - Password for Websense authentication
 - **target_url** - End point URL of Digital Analytics, already configured in Unica Campaign
 - On the Map context roots for Web modules page, set the Context Root to `proxy`.
 - When deployment is complete, access the ProxyServer application in a browser at `http://WebSphere_host:Port/proxy`.

You should receive a message: IBM OCM Secure Proxy Server V.x

Importing the Digital Analytics certificate when WebSphere does not have outbound access

Use this procedure when WebSphere does not have outbound access to the Digital Analytics server.

1. Retrieve the Digital certificate from the Digital Analytics site.

To retrieve the certificate, go to the Digital Analytics URL and click the lock icon in your browser's address field. Your browser opens a window where you can download the certificate.

2. Import the certificate into the WebSphere JVM using `java keytool`.

For example (line breaks added):

```
/keytool -import -file DA_Certificate.cer
-alias da_alias
-keystore WebSphere_JRE_home/lib/security/cacerts
```

Provide the password. The default keytool password is `changeit`.

3. In the WebSphere administrative console, add the following custom properties.
 - `javax.net.ssl.trustStore`: `WebSphere_JRE_home/lib/security/cacerts`
 - `javax.net.ssl.trustStorePassword`: `your_password`
 - `javax.net.ssl.trustStoreType`: `jks`

Importing the Digital Analytics certificate when WebSphere has outbound access

Use this procedure when WebSphere has outbound access to the Digital Analytics server.

1. In the WebSphere administrative console, expand **Security**, click **SSL certificate and key management**.
2. Under **Configuration settings**, click **Manage endpoint security configurations**.
3. Select the appropriate outbound configuration to navigate to the **(cell):..Node0xCCell:(node):..Node0x** management scope.
4. Under **Related Items**, click **Key stores and certificates** and click the **NodeDefaultTrustStore** (or the KeyStore you have used in WebSphere application server) key store.
5. Under **Additional Properties**, click **Signer certificates** and **Retrieve from port**.
 - a. In the **Host** field, enter the Digital Analytics server name.

For example, `export.coremetrics.com`.
 - b. In the **Port** field, enter `443`
 - c. In the **Alias** field, enter an alias name.
6. Click **Retrieve Signer Information** and verify that the certificate information is for a certificate that you can trust. .
7. Apply and save your configuration.

Next steps

After you install the custom proxy server and import the Digital Analytics certificate, your next steps are to enable single sign-on and configure integration between Digital Analytics and Unica Campaign.

To finish setting up your environment, perform the following procedures.

- Set up single sign on as described in the *Unica Platform Administrator's Guide*, in the chapter titled "Single sign-on between Unica and Digital Analytics."
- Set up integration as described in the *Unica Campaign Administrator's Guide*, in the chapter titled "Unica Campaign integration with other products."



Important: The integration procedure includes setting the `ServiceURL` configuration property under **Campaign | partitions | partition[n] | Coremetrics**. When you use the custom proxy, you must set this property to `http://WebSphere_host:Port/proxy`, and restart the Unica Platform web application.

Integration between Unica and Windows™ Active Directory

Unica Platform can be configured to integrate with Windows™ Active Directory server or another LDAP (Lightweight Directory Access Protocol) server. By integrating Unica 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 Unica 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

Unica Platform integration with Windows™ Active Directory provides the features described in this section.

Authentication with Active Directory integration

Unica applications query Unica Platform for user authorization information.

- Previous versions of Unica Platform supported NTLMv1 based Microsoft Windows Integrated login. With the arrival of Microsoft Windows 2008 Server and Microsoft Windows 7, the default minimum standard has changed and requires the NTLMv2 protocol. NTLMv2 is not natively supported by Unica Platform.

However, you can configure NTLMv2 authentication, so that users are authenticated to all Unica applications when they log in to the corporate network, and no password is required to log in to Unica applications. User authentication is based on their Windows™ login, bypassing the applications' login screens.

To configure NTLMv2 authentication, you perform the steps described in this chapter.

- If NTLMv2 authentication is not enabled, users must still log in on the Unica login screen, using their Windows™ credentials.

Managing internal and external users

When NTLMv2 authentication is enabled, all users are created and maintained in the Active Directory server. (You do not have the option of creating some users in Unica Platform, which are known as internal users in this guide). If you require the ability to create internal users, do not enable NTLMv2 authentication.

When integration is configured, you cannot add, modify, or delete the imported user accounts in Unica Platform. You must perform these management tasks on the LDAP side, and your changes are imported when synchronization occurs. If you modify imported user accounts in Unica Platform, users may encounter problems with authentication.

Any user accounts you delete on the LDAP side are not deleted from Unica Platform. You should disable these accounts manually in Unica Platform. It is safer to disable these deleted user accounts rather than deleting them, because users have folder ownership privileges in Unica Campaign, and if you delete a user account that owns a folder, objects in that folder will no longer be available.

Synchronization

When Unica is configured to integrate with an Active Directory server, users and groups are synchronized automatically at pre-defined intervals.

Automatic synchronization has limited functionality.

- Users deleted from the LDAP server are not deleted during automatic synchronization.

You can force a full synchronization of all users and groups by using the Synchronize function in the Users area of Unica. Alternatively, you can contact Services to request that they set a hidden configuration property that causes the automatic synchronization to perform a full synchronization.

Importing users based on groups or attributes

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

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

Group based import

Unica 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 Unica Platform imports users and groups from the server database, group memberships are not changed. To pick up these changes, you must perform a manual synchronization.

You can assign Unica privileges by mapping an Active Directory group to an Unica group. This mapping allows any new users added to the mapped Active Directory group to assume the privileges set for the corresponding Unica group.

A subgroup in Unica 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 Unica products, you have the option to control the users who are imported by specifying attributes. To achieve this, you would do the following during the configuration process.

1. Determine the string used in your Active Directory server for the attribute on which you want to filter.
2. Set the **Unica Platform | Security | LDAP synchronization | LDAP user reference attribute name** property to DN.

This indicates to Unica 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 Unica 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 Unica group most recently mapped to an Active Directory group is the one that the user belongs to. To determine which Active Directory group was most recently mapped, look at the LDAP group mappings displayed in the Configuration area. They are displayed in chronological order, with the most recent mapping listed last.

Special characters in login names

Only three special characters are allowed in login names: dot (.), underscore (_), and hyphen (-). If any other special characters (including spaces) are present in the login name of a user you plan to import into Unica 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).

Active Directory integration prerequisites

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

In addition, to implement NTLMv2 authentication, users accessing Unica applications must:

- Use a system running a supported Windows™ operating system.
- Log in as a member of the Windows™ Active Directory domain against which Unica is authenticating.
- Use a supported browser.

Configuration process roadmap: Active Directory integration

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

Table 38. Configuration process roadmap: Active Directory integration

Topic	Information
Obtaining required information (on page 131)	Obtain information about your Windows™ Active Directory server, which is required for integration with Unica.
Group membership, mapping, and application access (on page 132)	If you are using group based synchronization, identify or create the groups in Unica Platform to which you will map your Active Directory groups.
Storing directory server credentials in Unica Platform (on page 132)	If your directory server does not allow anonymous access (the most common configuration), configure an Unica user account to hold a directory server administrator user name and password.
<ul style="list-style-type: none"> • Setting LDAP login method connection properties in Unica (on page 133) • Setting LDAP synchronization properties (on page 134) • Setting user attributes map properties (on page 135) • Mapping LDAP groups to Unica groups (on page 136) 	Configure the Unica Platform for integration by setting values on the Configuration page.
Testing synchronization (on page 136)	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 137)	Set up administrator access to Unica Platform, required when NTLMv2 authentication is enabled.
Setting the security mode to enable NTLMv2 authentication (on page 137)	Set the security mode values on the Configuration page.
Configuring Internet Explorer (on page)	Set a custom security level in every instance of Internet Explorer that is used to access Unica. This is required with NTLMv2 authentication, to prevent users from being presented with the Unica login screen.
Restarting the web application server (on page 137)	This step is required to ensure that all of your changes are applied.
Testing login as an Active Directory user (on page 137)	Verify that you can log in to Unica as an Active Directory user.

Obtaining required information

Obtain the required information about the directory server with which you want to integrate. You use this information during the configuration process, to store directory server credentials and to set configuration property values.

Obtain the following information.

- Obtain the server host name and port.
- Identify a user who has search permissions on the directory server, and gather the following information about the user.
 - login name
 - password
 - Distinguished Name (DN)
- Obtain the following for the directory server.
 - Fully qualified host name or IP address
 - The port on which server listens
- Determine the string that your directory server uses for the user attribute in the Group object. Typically, this value is `uniqueMember` in LDAP servers and `member` in Windows™ Active Directory servers. You should verify this on your directory server.
- Obtain the following required user attributes.
 - Determine the string that your directory server uses for the user login attribute. This string is always required. Typically, this value is `uid` in LDAP servers and `sAMAccountName` in Windows™ Active Directory servers. Verify this string on your directory server.
 - Only if Unica Campaign is installed in a UNIX™ environment, determine the string that your directory server uses for the alternate login attribute.
- If you are using attribute based synchronization, obtain the strings used for the attributes (one or more) that you want to use for this purpose.
- If you want Unica 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 Unica, 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 Unica Platform groups, follow the guidelines described here.

- Identify or create the directory server groups whose members you want to import into the Unica Platform. When these groups are mapped to Unica Platform groups, members of these groups are automatically created as Unica users.

Members of your directory server's subgroups are not imported automatically. To import users from subgroups, you must map the subgroups to Unica 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 Unica 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 Unica Platform

If your directory server does not allow anonymous access, you must configure an Unica user account to hold the user name and password of a directory server user, as described in the following procedure.

1. Log in to Unica as a user with Admin access.
2. Select or create an Unica 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 `Unica Platform user for LDAP credentials configuration` property to the user name for this Unica user account. The default value of this property is `asm_admin`, a user that exists in every new Unica Platform installation. You can use the `asm_admin` account to hold the directory server credentials.
 - The user name of this Unica user account must not match the user name of any directory server user.
3. Add a data source for this Unica user account to store the credentials that Unica Platform uses to connect with the LDAP server. Follow these guidelines.

Table 39. Data source fields for storing credentials

Field	Guideline
Data Source Name	<p>You can enter any name, but note that in a later step, the value of the <code>Data source for LDAP credentials configuration</code> 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>.</p>
Data Source Login	<p>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 Unica. The DN resembles the following:</p> <pre>uidcn=user1,ou=someGroup,dc=systemName,dc=com</pre> <p>Alternatively, you can use the root user account that has access to all groups on your LDAP server. The default root user and how to specify this user for the supported directory servers are as follows.</p> <ul style="list-style-type: none"> ◦ The root user for Active Directory Server is Administrator. You can specify this user as follows. <pre>domain\ldap_admin_username</pre> ◦ The root user for Oracle Directory Server is Directory Manager. You can specify this user as follows. <pre>cn=Directory Manager</pre> ◦ The root user for IBM Security Directory Server is root. You can specify this user as follows. <pre>cn=root</pre>
Data Source Password	<p>Enter the password of the administrative user whose login name you entered in the Data Source Login field.</p>

Setting LDAP login method connection properties in Unica

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

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

See the related reference for details on how to set each property.

- LDAP server host name
- LDAP server port
- User search filter
- Use credentials stored in Unica Platform
- Unica 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.

1. Click **Settings > Configuration** and navigate to the **Platform | Security | LDAP Synchronization** category.
2. Set values of the following configuration properties in the **LDAP properties** section.

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

- LDAP sync enabled
- LDAP sync interval
- LDAP sync delay
- LDAP sync timeout
- LDAP sync scope
- LDAP provider URL
- Require SSL for LDAP connection (optional)
- LDAP config Unica Platform group delimiter
- LDAP reference config delimiter

- Unica Platform user for LDAP credentials
- Data source for LDAP credentials
- LDAP user reference attribute name
- LDAP BaseDN periodic search disabled
- 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.

1. Click **Settings > Configuration** and navigate to the **Platform | Security | LDAP Synchronization** category.
2. Set values in the **User attributes map** section to map the listed Unica user attributes to the user attributes in your directory server.

If you are using group based synchronization, the only property you are required to map is `User login`. Typically, this value is `uid` in LDAP servers and `sAMAccountName` in Windows™ Active Directory servers. Use the value you verified as described in "Obtaining required information."

If you are using attribute based synchronization, map the attributes on which you want to search.

Note the following.

- The properties that you map here are replaced for the imported users each time Unica Platform synchronizes with your directory server.
- Unica 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 Unica Platform system tables, as shown in the following table, the attribute value is truncated to fit.

Table 40. Number of characters allowed for user attributes

Attribute	Allowed length
User login (required)	256
First name	128
Last name	128
User title	128
Department	128

Attribute	Allowed length
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 Unica groups

Users who belong to the directory server groups you map here are imported and made members of the Unica Platform group or groups specified here.



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

1. Click **Settings > Configuration** and navigate to the **Unica | Unica Platform | Security | LDAP Synchronization | LDAP reference to Unica Platform group map** category.
2. For each directory server group you want to map to a Unica Platform group, create an **LDAP reference to Unica Platform group** category by selecting the *(LDAP reference to Unica Platform group map)* template. Set the following properties.
 - New category name
 - LDAP reference map
 - Unica Platform group

For example, the following values map the LDAP`MarketingPlatformUsers` group to the Unica Platform `marketingopsUsers` and `campaignUsers` groups (`FILTER` is omitted).

- LDAP reference: `cn=MarketingPlatformUsers,cn=Users, dc=myCompany,dc=com`
- Unica Platform group: `marketingopsUsers;campaignUsers`

Testing synchronization

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

1. Log in to Unica as an Unica 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 Unica Platform group memberships match the expected mapping to directory server groups.

Setting up an Active Directory user with PlatformAdminRole permissions

When NTLMv2 authentication is enabled, you can not log in to Unica as `platform_admin`, so you must perform the following procedure in order to have administrator access to Unica Platform.

1. Log in to Unica as an internal user (a user created in Unica Platform rather than a user imported from Active Directory). This must be a user with PlatformAdminRole permissions in the Unica Platform.
2. Create a Unica 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 enable NTLMv2 authentication

Only if you want to enable NTLMv2 authentication, set configuration properties as described in this procedure.

Configure NTLMv2 authentication.

Click **Settings > Configuration** and set configuration properties as shown in the following table.

Table 41. Configuration property values for NTLMv2

Property	Value
Platform Security Login method	Select the <code>web access control</code> option.
Platform Security Login method details Web access control Web access control header variable	Enter the variable name as specified in the re-write rules.
Platform Security Login method details Web access control User-name pattern	Enter <code>\w*</code>
General Navigation Platform URL	Enter the IIS site URL.

Restarting the web application server

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

Testing login as an Active Directory user

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

1. Log in to Windows™ as an Active Directory user who is a member of an Active Directory group mapped to a Unica Platform group that has been assigned a role in the Unica Platform.
2. Point your browser to the Unica URL.

If you have enabled NTLMv2 authentication, you should not see the Unica login screen, and you should be allowed to access the Unica user interface.

If you have not enabled NTLMv2 authentication, you should be able to log in with your Windows credentials.

If you cannot log in, see [restoreAccess \(on page 253\)](#).

Integration between Unica and LDAP servers

Unica Platform can be configured to integrate with Windows™ Active Directory server or another LDAP (Lightweight Directory Access Protocol) server. By integrating Unica 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 Unica 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

Unica Platform integration with LDAP provides the features described in this section.

Authentication with LDAP integration

Unica applications query Unica Platform for user authorization information. When LDAP integration is implemented, users enter their valid LDAP user name and password for authentication to Unica applications.

Managing internal and external users

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

Any user accounts you delete on the LDAP side are not deleted from Unica Platform. You should disable these accounts manually in Unica Platform. It is safer to disable these deleted user accounts rather than deleting them, because users have folder ownership privileges in Unica Campaign, and if you delete a user account that owns a folder, objects in that folder will no longer be available.

Synchronization

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

Automatic synchronization has limited functionality.

- Users deleted from the LDAP server are not deleted during automatic synchronization.

You can force a full synchronization of all users and groups by using the Synchronize function in the Users area of Unica. Alternatively, you can contact Services to request that they set a hidden configuration property that causes the automatic synchronization to perform a full synchronization.

Importing users based on groups or attributes

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

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

Group based import

Unica 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 Unica Platform imports users and groups from the server database, group memberships are not changed. To pick up these changes, you must perform a manual synchronization.



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

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

A subgroup in Unica 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 Unica products, you have the option to control the users who are imported by specifying attributes. To achieve this, you would do the following during the LDAP configuration process.

1. Determine the string used in your LDAP server for the attribute on which you want to filter.
2. Set the **Platform | Security | LDAP synchronization | LDAP user reference attribute name** property to DN.

This indicates to Unica 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 Unica 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 Unica 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

Unica supports two types of user accounts and groups.

- **Internal** - User accounts and groups that are created within Unica using the Unica security user interface. These users are authenticated through Unica Platform.
- **External** - User accounts and groups that are imported into Unica through synchronization with a supported LDAP server. This synchronization occurs only if Unica 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 Unica applications without adding them to your LDAP server as full corporate users.

Using this hybrid authentication model requires more maintenance than a pure LDAP authentication model does.

Special characters in login names

Only three special characters are allowed in login names: dot (.), underscore (_), and hyphen (-). If any other special characters (including spaces) are present in the login name of a user you plan to import into Unica 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).

LDAP integration prerequisites

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

Table 42. Configuration process roadmap: LDAP integration

Topic	Information
Obtaining required information (on page 131)	Obtain information about your LDAP server, which is needed for integration with Unica.
Group membership, mapping, and application access (on page 132)	If you are using group based synchronization, identify or create the groups in Unica Platform to which you will map your LDAP groups.
Storing directory server credentials in Unica Platform (on page 132)	If your directory server does not allow anonymous access (the most common configuration), configure an Unica user account to hold a directory server administrator user name and password.
<ul style="list-style-type: none"> • Setting LDAP login method connection properties in Unica (on page 133) • Setting LDAP synchronization properties (on page 134) • Setting user attributes map properties (on page 135) • Mapping LDAP groups to Unica groups (on page 136) 	Configure the Unica Platform for integration by setting values on the Configuration page.
Testing synchronization (on page 136)	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 147)	Set the security mode values on the Configuration page.
Restarting the web application server (on page 137)	This step is required to ensure that all of your changes are applied.
Testing login as an LDAP user (on page 148)	Verify that you can log in to Unica as an LDAP user.

Obtaining required information

Obtain the required information about the directory server with which you want to integrate. You use this information during the configuration process, to store directory server credentials and to set configuration property values.

Obtain the following information.

- Obtain the server host name and port.
- Identify a user who has search permissions on the directory server, and gather the following information about the user.

- login name
- password
- Distinguished Name (DN)
- Obtain the following for the directory server.
 - Fully qualified host name or IP address
 - The port on which server listens
- Determine the string that your directory server uses for the user attribute in the Group object. Typically, this value is `uniqueMember` in LDAP servers and `member` in Windows™ Active Directory servers. You should verify this on your directory server.
- Obtain the following required user attributes.
 - Determine the string that your directory server uses for the user login attribute. This string is always required. Typically, this value is `uid` in LDAP servers and `sAMAccountName` in Windows™ Active Directory servers. Verify this string on your directory server.
 - Only if Unica Campaign is installed in a UNIX™ environment, determine the string that your directory server uses for the alternate login attribute.
- If you are using attribute based synchronization, obtain the strings used for the attributes (one or more) that you want to use for this purpose.
- If you want Unica 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 Unica, 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 Unica Platform groups, follow the guidelines described here.

- Identify or create the directory server groups whose members you want to import into the Unica Platform. When these groups are mapped to Unica Platform groups, members of these groups are automatically created as Unica users.

Members of your directory server's subgroups are not imported automatically. To import users from subgroups, you must map the subgroups to Unica 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 Unica 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 Unica Platform

If your directory server does not allow anonymous access, you must configure an Unica user account to hold the user name and password of a directory server user, as described in the following procedure.

1. Log in to Unica as a user with Admin access.
2. Select or create an Unica 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 `Unica Platform user for LDAP credentials configuration` property to the user name for this Unica user account. The default value of this property is `asm_admin`, a user that exists in every new Unica Platform installation. You can use the `asm_admin` account to hold the directory server credentials.
 - The user name of this Unica user account must not match the user name of any directory server user.
3. Add a data source for this Unica user account to store the credentials that Unica Platform uses to connect with the LDAP server. Follow these guidelines.

Table 43. Data source fields for storing credentials

Field	Guideline
Data Source Name	<p>You can enter any name, but note that in a later step, the value of the <code>Data source for LDAP credentials</code> 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>.</p>
Data Source Login	<p>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 Unica. The DN resembles the following:</p> <pre>uidcn=user1,ou=someGroup,dc=systemName,dc=com</pre> <p>Alternatively, you can use the root user account that has access to all groups on your LDAP server. The default root user and how to specify this user for the supported directory servers are as follows.</p> <ul style="list-style-type: none"> ◦ The root user for Active Directory Server is Administrator. You can specify this user as follows. <pre>domain\ldap_admin_username</pre> ◦ The root user for Oracle Directory Server is Directory Manager. You can specify this user as follows. <pre>cn=Directory Manager</pre> ◦ The root user for IBM Security Directory Server is root. You can specify this user as follows. <pre>cn=root</pre>
Data Source Password	<p>Enter the password of the administrative user whose login name you entered in the Data Source Login field.</p>

Setting LDAP login method connection properties in Unica

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

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

See the related reference for details on how to set each property.

- LDAP server host name

- LDAP server port

- User search filter
- Use credentials stored in Unica Platform
- Unica 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.

1. Click **Settings > Configuration** and navigate to the **Platform | Security | LDAP Synchronization** category.
2. Set values of the following configuration properties in the **LDAP properties** section.

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

- LDAP sync enabled
- LDAP sync interval
- LDAP sync delay
- LDAP sync timeout
- LDAP sync scope
- LDAP provider URL
- Require SSL for LDAP connection (optional)
- LDAP config Unica Platform group delimiter
- LDAP reference config delimiter
- Unica Platform user for LDAP credentials
- Data source for LDAP credentials
- LDAP user reference attribute name
- LDAP BaseDN periodic search disabled

- 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.

1. Click **Settings > Configuration** and navigate to the **Platform | Security | LDAP Synchronization** category.
2. Set values in the **User attributes map** section to map the listed Unica user attributes to the user attributes in your directory server.

If you are using group based synchronization, the only property you are required to map is `User login`. Typically, this value is `uid` in LDAP servers and `sAMAccountName` in Windows™ Active Directory servers. Use the value you verified as described in "Obtaining required information."

If you are using attribute based synchronization, map the attributes on which you want to search.

Note the following.

- The properties that you map here are replaced for the imported users each time Unica Platform synchronizes with your directory server.
- Unica 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 Unica Platform system tables, as shown in the following table, the attribute value is truncated to fit.

Table 44. Number of characters allowed for user attributes

Attribute	Allowed length
User login (required)	256
First name	128
Last name	128
User title	128
Department	128
Company	128
Country	128
User email	128
Address 1	128

Attribute	Allowed length
Work phone	20
Mobile phone	20
Home phone	20
Alternate login (required on UNIX™)	256

Mapping LDAP groups to Unica groups

Users who belong to the directory server groups you map here are imported and made members of the Unica Platform group or groups specified here.



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

1. Click **Settings > Configuration** and navigate to the **Unica | Unica Platform | Security | LDAP Synchronization | LDAP reference to Unica Platform group map** category.
2. For each directory server group you want to map to a Unica Platform group, create an **LDAP reference to Unica Platform group** category by selecting the *(LDAP reference to Unica Platform group map)* template. Set the following properties.

- New category name
- LDAP reference map
- Unica Platform group

For example, the following values map the LDAP`MarketingPlatformUsers` group to the Unica Platform `marketingopsUsers` and `campaignUsers` groups (`FILTER` is omitted).

- LDAP reference: `cn=MarketingPlatformUsers,cn=Users,dc=myCompany,dc=com`
- Unica Platform group: `marketingopsUsers;campaignUsers`

Testing synchronization

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

1. Log in to Unica as an Unica 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 Unica 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 Unica applications.

1. Log in to Unica, click **Settings > Configuration**, and navigate to **Unica 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 Unica as an LDAP user who is a member of an LDAP group mapped to a Unica Platform group that has been assigned access to Unica Platform.

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

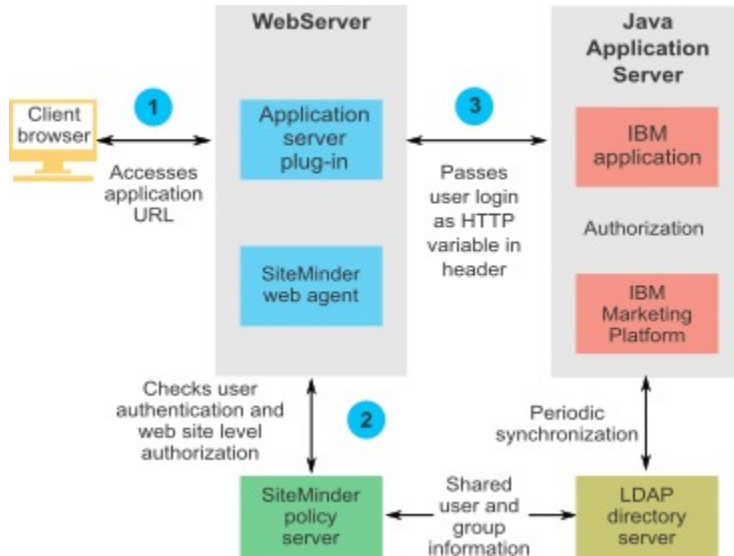
Authorization

Unica applications query Unica Platform for user authorization information. Unica Platform imports groups and their users from the LDAP database through a periodic synchronization task that automatically retrieves information from the LDAP server. When Unica 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 Unica are referencing a consistent set of users.

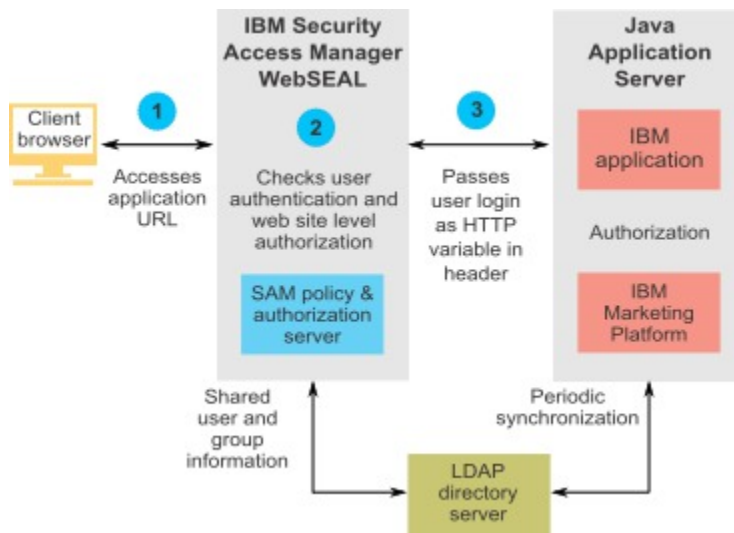
Additional authorization controls, including control over the application URLs to which users have access, are also available through most web access control systems.

Web access control integration diagrams

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



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



About context roots

You must unprotect URLs in your web access control system to enable various features in Unica products. To perform this task, you need to include the product context roots in the URLs.

The following table provides a list of the default context roots for the Unica products mentioned in this chapter. Your installation might use non-default context roots, but typically most installations accept the default.

The examples in this chapter use the default context roots. If your environment uses a non-standard context root, you must change the context root shown in the example URLs to the context root used in your environment.

Table 45. Context roots for Unica products

Product	Context root
Unica Platform	unica
Unica Campaign	Campaign
Unica Optimize	Campaign/optimize
Unica Plan	plan
Unica Collaborate	collaborate
Unica Interact	Campaign/interact

SiteMinder integration prerequisites

The following prerequisites must be met to integrate Unica 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 Unica application.
- The Unica property **Web access control header variable** must be set to the name of the variable that SiteMinder uses for login names.

The default name for the SiteMinder login name variable is `sm_user`.

- The SiteMinder policy server must be configured to use LDAP as its repository for storing group members and user properties.
- The Unica application URLs provided by the web server hosting SiteMinder and the Java™ application server hosting the Unica application must refer to the same path.
- The web server hosting SiteMinder must be configured to redirect requests to the Unica application URL on the Java™ application server.
- All users who need to access Unica applications must be granted access in SiteMinder to the Unica 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 Unica products.

Configuring SiteMinder for Unica products

Unprotect objects in SiteMinder as described in this procedure to enable correct functioning of your Unica products.

1. Log in to the **Administer Policy Server** area of SiteMinder and click **Domains**.
2. Select the realm that applies to your installations, right-click **unprotecturl**, and select **Properties of Realm**.
3. For each of the applicable URLs as described in the following table, enter the URL in the **Resource Filter** text box, and under **Default Resource Protection**, select **Unprotected**.

Table 46. Un-protected objects required for Unica products

Product or feature	Objects
Unica Campaign	<ul style="list-style-type: none"> ◦ <code>/Campaign/services/CampaignServices30Service</code> ◦ <code>/Campaign/api/campaign/rest</code> ◦ <code>/Campaign/FlowchartNotifyScheduler</code> ◦ <code>/Campaign/OperationMonitor</code> ◦ <code>http://host:port/Campaign/api/campaign/rest/deepsearch/partition</code> <p>Replace partition with the partition name.</p> <p>The following apply when integration with Engage is implemented.</p> <p>In the following URLs, replace partition with the partition name.</p> <ul style="list-style-type: none"> ◦ <code>http://host:port/Campaign/jsp/engage/engageHome.jsp</code> ◦ <code>http://host:port/Campaign/api/campaign/rest/engage/offers</code> ◦ <code>http://host:port/Campaign/api/campaign/rest/engage/offer</code> ◦ <code>http://host:port/Campaign/servlet/EngageUpload</code> ◦ <code>http://host:port/Campaign/api/campaign/rest/engageim-portlist/partition</code> ◦ <code>http://host:port/Campaign/api/campaign/rest/engageim-portlist/partition/jobid</code> <p>This URL is for checking the status of an import job. Replace jobid with your job ID.</p> <ul style="list-style-type: none"> ◦ <code>http://host:port/Campaign/api/campaign/rest/engageim-portlist/partition/schedule</code> ◦ <code>http://host:port/Campaign/api/campaign/rest/engageim-portlist/partition/channel/schedule</code> <p>This URL is for sending push or SMS messages. Channel is either <code>sms</code> or <code>push</code>.</p>
Unica Journey	<ul style="list-style-type: none"> ◦ <code>/journey/api/platformlogin</code> ◦ <code>/journey/api/datadefinitions</code> ◦ <code>/journey/api/entrysources</code> ◦ <code>/journey/api/journeys</code> ◦ <code>/journey/api/folders</code> ◦ <code>/journey/api/permissions</code> ◦ <code>/unica/api/manager/authentication/login</code> ◦ <code>/unica/api/manager/user/user-details</code> ◦ <code>/unica/api/manager/configuration/get</code> ◦ <code>/unica/api/manager/policy/roles-permissions</code>

Product or feature	Objects
	<ul style="list-style-type: none"> ◦ /unica/api/manager/license/7 ◦ /unica/api/manager/datasource ◦ /journey/api/thirdpartylogin
Unica Collaborate	<ul style="list-style-type: none"> ◦ /collaborate/affiniumcollaborate.jsp ◦ /collaborate/services/CollaborateIntegrationServices1.0 ◦ /collaborate/flowchartRunNotifyServlet ◦ /collaborate/js/js_messages.jsp ◦ /collaborate/js/format_symbols.jsp ◦ /collaborate/alertsService
Unica Deliver	/Campaign/deliver/eventSinkServlet
Unica Interact	<ul style="list-style-type: none"> ◦ /Campaign/interact/saveFlowchartAction.udo ◦ /Campaign/interact/flowchartEventPatterns.udo ◦ /Campaign/interact/testRunFlowchart.udo ◦ /Campaign/interact/getProfileDataAction.udo ◦ /Campaign/interact/manageIPB.udo ◦ /Campaign/interact/flowchartRTAttrs.udo ◦ /Campaign/initOfferListResolution.udo ◦ /Campaign/getOfferListResolutionStatus.udo
Unica Plan	<ul style="list-style-type: none"> ◦ /plan/errorPage.jsp ◦ /plan/alertsService ◦ /plan/services ◦ /plan/services/collabService ◦ /plan/services/PlanIntegrationServices/1.0 ◦ /plan/affiniumplan.jsp ◦ /plan/invalid_user.jsp ◦ /plan/js/js_messages.jsp ◦ /plan/js/format_symbols.jsp ◦ /unica/servlet/AJAXProxy ◦ /plan/api/plan ◦ /plan/api/plan/flowchartApproval/flowchartApproval/validate
Unica Optimize	<ul style="list-style-type: none"> ◦ /Campaign/optimize/ext_runOptimizeSession.do ◦ /Campaign/optimize/ext_optimizeSessionProgress.do ◦ /Campaign/optimize/ext_doLogout.do
IBM SPSS Modeler Advantage Enterprise Marketing Management Edition	/unica/rest/spssUser

Product or feature	Objects
Unica Platform data filters	<code>/unica/servlet/DataFiltering</code>
Unica Platform	<ul style="list-style-type: none"> ◦ <code>/unica/api/manager/rest/v2/</code> ◦ <code>/unica/api/manager/rest/v2/users</code> ◦ <code>/unica/api/manager/rest/v2/roles</code> ◦ <code>/unica/api/manager/rest/v2/utilities</code>
Unica notifications	<ul style="list-style-type: none"> ◦ <code>unica/servlet/alertAJAXProxy</code> ◦ <code>unica/notification/alertsCount</code>
Unica Scheduler	<code>/unica/servlet/SchedulerAPIServlet</code>

Enabling single logouts with SiteMinder

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

1. Log in to the **Administer Policy Server** area of SiteMinder and set the `logoffUri` property to the URI of the Unica logout page.

For example: `/sm_realm/unica/j_spring_security_logout` where `sm_realm` is the SiteMinder security realm and `unica` is the Unica Platform context root.

2. Unprotect the Unica logout page, `/unica/jsp/frameworklogout.jsp` to ensure that SiteMinder does not force the user to sign in again to view the logout page.

Enabling custom logouts with SiteMinder

To enable custom logouts with SiteMinder, set `unica.sm.logouturl` under `Affinium|suite|security|loginModes|siteMinderPartitionLogin` by performing the following steps.

1. Get the configuration ID of the Platform's hidden configuration `unica.sm.logouturl` using the following query.

```
select ID from USM_CONFIGURATION where INTERNAL_NAME = 'unica.sm.logouturl'
```

2. Update the value of configuration element `unica.sm.logouturl`:

```
update USM_CONFIGURATION_VALUES set STRING_VALUE='<custom logout url>' where CONFIGURATION_ID=<ID
obtained from above query>
```

IBM Security Access Manager integration prerequisites

The following prerequisites must be met to integrate Unica with IBM Security Access Manager.

- The IBM Security Access Manager WebSEAL junction must be configured to pass the user name (Short, not Full DN) as the HTTP variable in the URL request to the Unica application.
- The Unica property `Web access control header variable` must be set to the name of the variable that Security Access Manager uses for login names.

The default name for the Security Access Manager login name variable is `iv-user`.

- The IBM Security Access Manager policy server must be configured to use LDAP as its repository for storing group members and user attributes.
- The Unica application URLs defined by a WebSEAL junction and the Java™ application server hosting the Unica application must refer to the same path.
- All users who need to access Unica 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 Unica Platform is deployed must be attached to this ACL.
- To ignore basic authentication on ISAM setup, you must set `Ignore HTTP Basic Authentication header`. Navigate to **Junction Management** -> **<Edit Junction>** -> **Identity Tab** and select **Ignore** for HTTP Basic Authentication Header.



Note: When users log out of an Unica application, they are not automatically logged out of IBM Security Access Manager. They must close their browser after they log out of an Unica application to log out of IBM Security Access Manager.

Configuring IBM Security Access Manager for Unica products

Unprotect objects in IBM Security Access Manager as described in this procedure to enable correct functioning of your Unica products.

1. Use Web Portal Manager to log in to the domain as a domain administrator.
2. Click **ACL > Create ACL**, complete the **Name** and **Description** fields, and click **Apply**.
3. Click **ACL > List ACL**, and from the Manage ACLs page, click the link for your ACL policy.
4. From the ACL Properties page, click **Create**, and create two entries for your ACL, as follows.
 - For the first entry, set the entry type to **unauthenticated** and grant **Trx - Traverse, read, delete and execute** permissions.
 - For the second entry, set the entry type to **Any-other** and grant **Trx - Traverse, read, delete and execute** permissions.
5. On the ACL Properties page of the ACL, on the Attach tab, attach un-protected objects, as required for your product installations.

Use the complete path in IBM Security Access Manager, starting from WebSEAL.

Table 47. Un-protected objects required for Unica products

Product or feature	Objects
Unica Campaign	<ul style="list-style-type: none"> ◦ /WebSEAL junction/Campaign/optimize/ext_runOptimizeSession.do ◦ /WebSEAL junction/Campaign/optimize/ext_optimizeSessionProgress.do ◦ /WebSEAL junction/Campaign/optimize/ext_doLogout.do ◦ /WebSEAL junction/Campaign/interact/flowchartEventPatterns.udo ◦ /WebSEAL junction/Campaign/interact/saveFlowchartAction.udo ◦ /WebSEAL junction/Campaign/interact/testRunFlowchart.udo ◦ /WebSEAL junction/Campaign/interact/getProfileDataAction.udo ◦ /WebSEAL junction/Campaign/interact/manageIPB.udo ◦ /WebSEAL junction/Campaign/servlet/EngageUpload ◦ /WebSEAL junction/Campaign/api/campaign/rest/engageimportlist/partition ◦ /WebSEAL junction/Campaign/api/campaign/rest/engageimportlist/partition/schedule ◦ /WebSEAL junction/Campaign/api/campaign/rest/engageimportlist/partition/channel/schedule ◦ /WebSEAL junction/Campaign/interact/interactiveChannelSimulator.do ◦ /WebSEAL junction/Campaign/interact/interactiveChannelOfferMapping.do ◦ /WebSEAL junction/Campaign/services/CampaignServices30Service ◦ /WebSEAL junction/Campaign/FlowchartNotifyScheduler ◦ /WebSEAL junction/Campaign/OperationMonitor ◦ /WebSEAL junction/Campaign/initOfferListResolution.udo ◦ /WebSEAL junction/Campaign/getOfferListResolutionStatus.udo ◦ /WebSEAL junction/Campaign/moveCampaignsSubmit.do ◦ /WebSEAL junction/Campaign/interact/interactiveChannelStrategy.do ◦ /WebSEAL junction/Campaign/api/interact/rest ◦ /WebSEAL junction/Campaign/api/campaign/rest ◦ /WebSEAL junction/Campaign/interact/flowchartRTAttrs.udo ◦ /WebSEAL junction/Campaign/api/campaign/rest/ deepsearch/partition ◦ /WebSEAL junction/Campaign/api/interact/rest/v2 ◦ /WebSEAL junction/Campaign/api/interact/rest/v2/channels?page=0&size=1000 ◦ /WebSEAL junction/journey/api/campaign ◦ WebSEAL junction/Campaign/services/CampaignServices30Service ◦ WebSEAL junction/Campaign/api/campaign/rest ◦ WebSEAL junction/Campaign/FlowchartNotifyScheduler ◦ WebSEAL junction/Campaign/initOfferListResolution.udo ◦ WebSEAL junction/Campaign/getOfferListResolutionStatus.udo

Product or feature	Objects
	<ul style="list-style-type: none"> ◦ <code>WebSEAL junction/Campaign/OperationMonitor</code> ◦ <code>WebSEAL junction/Campaign/api/campaign/rest</code> ◦ <code>http://host:port/Campaign/api/campaign/rest/deepsearch/partition</code> <p>Replace partition with the partition name.</p> <p>The following apply when integration with Engage is implemented.</p> <p>In the following URLs, replace partition with the partition name.</p> <ul style="list-style-type: none"> ◦ <code>http://host:port/Campaign/jsp/engage/engageHome.jsp</code> ◦ <code>http://host:port/Campaign/api/campaign/rest/engage/offers</code> ◦ <code>http://host:port/Campaign/api/campaign/rest/engage/offer</code> ◦ <code>http://host:port/Campaign/servlet/EngageUpload</code> ◦ <code>http://host:port/Campaign/api/campaign/rest/engageimportlist</code> ◦ <code>http://host:port/Campaign/api/campaign/rest/engageimportlist/partition</code> ◦ <code>http://host:port/Campaign/api/campaign/rest/engageimportlist/partition/jobid</code> <p>This URL is for checking the status of an import job. Replace jobid with your job ID.</p> <ul style="list-style-type: none"> ◦ <code>http://host:port/Campaign/api/campaign/rest/engageimportlist/partition/schedule</code> ◦ <code>http://host:port/Campaign/api/campaign/rest/engageimportlist/partition/channel/schedule</code> <p>This URL is for sending push or SMS messages. Channel is either <code>sms</code> or <code>push</code>.</p>
Unica Collaborate	<ul style="list-style-type: none"> ◦ <code>WebSEAL junction/collaborate/affiniumcollaborate.jsp</code> ◦ <code>WebSEAL junction/collaborate/services/CollaborateIntegrationServices1.0</code> ◦ <code>WebSEAL junction/collaborate/flowchartRunNotifyServlet</code> ◦ <code>WebSEAL junction/collaborate/js/js_messages.jsp</code> ◦ <code>WebSEAL junction/collaborate/js/format_symbols.jsp</code> ◦ <code>WebSEAL junction/collaborate/alertsService</code>
Unica Journey	<ul style="list-style-type: none"> ◦ <code>/WebSEAL/<instance name>/<junction name>/journey/api/platformlogin</code> ◦ <code>/WebSEAL/<instance name>/<junction name>/journey/api/datadefinitions</code> ◦ <code>/WebSEAL/<instance name>/<junction name>/journey/api/entrysources</code> ◦ <code>/WebSEAL/<instance name>/<junction name>/journey/api/journeys</code> ◦ <code>/WebSEAL/<instance name>/<junction name>/journey/api/folders</code>

Product or feature	Objects
	<ul style="list-style-type: none"> ◦ /WebSEAL/<instance name>/<junction name>/journey/api/permissions ◦ /WebSEAL/<instance name>/<junction name>/unica/api/manager/authentication/login ◦ /WebSEAL/<instance name>/<junction name>/unica/api/manager/user/user-details ◦ /WebSEAL/<instance name>/<junction name>/unica/api/manager/configuration/get ◦ /WebSEAL/<instance name>/<junction name>/unica/api/manager/policy/roles-permissions ◦ /WebSEAL/<instance name>/<junction name>/unica/api/manager/license/7 ◦ /WebSEAL/<instance name>/<junction name>/unica/api/manager/data-source ◦ /WebSEAL/<instance name>/<junction name>/journey/api/thirdpartylogin
Unica Deliver	<p><i>WebSEAL junction/Campaign/deliver/eventSinkServlet</i></p>
Unica Interact	<ul style="list-style-type: none"> ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/interact/flowchart-EventPatterns.udo ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/interact/saveFlowchartAction.udo ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/interact/testRunFlowchart.udo ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/interact/getProfileDataAction.udo ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/interact/manageIPB.udo ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/initOfferListResolution.udo ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/getOfferListResolutionStatus.udo ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/interactiveChannelOfferMapping.do ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/interactiveChannelStrategy.do ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/interact/interactiveChannelOfferMapping.do ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/FlowchartNotifyScheduler

Product or feature	Objects
	<ul style="list-style-type: none"> ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/OperationMonitor ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/initOfferListResolution.udo ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/getOfferListResolutionStatus.udo ◦ /WebSEAL/<instancename>/<junctionname>/interact/servlet/InteractJSService ◦ /WebSEAL/<instancename>/<junctionname>/interact/servlet/RestServlet ◦ /WebSEAL/<instancename>/<junctionname>/interact/services/InteractService ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/api/campaign/rest ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/moveCampaignsSubmit.do ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/interact/flowchart-RTAttrs.udo ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/interact/interactiveChannelStrategy.do ◦ /WebSEAL/<instancename>/<junctionname>/Campaign/api/interact/rest ◦ /WebSEAL/<instancename>/<junctionname>/interact/InteractDT/api/interact/rest/v2/flowcharts/action ◦ /WebSEAL/<instancename>/<junctionname>/interact/InteractDT/api/interact/rest/v2/flowcharts/rtaUsage ◦ /WebSEAL/<instancename>/<junctionname>/interact/InteractDT/api/interact/rest/v2/flowcharts/testRun ◦ /WebSEAL/<instancename>/<junctionname>/interact/InteractDT/api/interact/rest/v2/flowcharts/profile ◦ /WebSEAL/<instancename>/<junctionname>/interact/InteractDT/api/interact/rest/v2/flowcharts/intProcessBox/management ◦ /WebSEAL/<instancename>/<junctionname>/interact/InteractDT/api/interact/rest/v2/flowcharts/eventPatterns ◦ /WebSEAL/<instancename>/<junctionname>/interact/InteractDT/api/interact/rest/v2/flowcharts/rtAttrs ◦ /WebSEAL/<instancename>/<junctionname>/interact/InteractDT/api/interact/rest/v2/flowcharts/removeLock ◦ /WebSEAL/<instancename>/<junctionname>/interact/InteractDT/api/interact/rest/v2/intFlowcharts ◦ /WebSEAL/<instancename>/<junctionname>/interact/InteractDT/api/interact/rest/v2/strategies ◦ /WebSEAL/<instancename>/<junctionname>/interact/InteractDT/api/interact/rest/v2/interactiveChannels

Product or feature	Objects
	<ul style="list-style-type: none"> ◦ /WebSEAL/<instancename>/<junctionname>/interact/InteractDT/api/interact/rest/v2/flowcharts ◦ /WebSEAL/<instancename>/<junctionname>/interact/InteractDT/api/interact/rest/v2/channels ◦ /WebSEAL/<instancename>/<junctionname>/interact/InteractDT/api/interact/rest/v2/rtAttrs
Unica Plan	<ul style="list-style-type: none"> ◦ WebSEAL junction/plan/services ◦ WebSEAL junction/plan/errorPage.jsp ◦ WebSEAL junction/plan/alertsService ◦ WebSEAL junction/plan/services/collabService ◦ WebSEAL junction/plan/services/PlanIntegrationServices/1.0 ◦ WebSEAL junction/plan/affiniumplan.jsp ◦ WebSEAL junction/plan/invalid_user.jsp ◦ WebSEAL junction/plan/js/js_messages.jsp ◦ WebSEAL junction/plan/js/format_symbols.jsp ◦ WebSEAL junction/unica/servlet/AJAXProxy ◦ WebSEAL junction//plan/api/plan/flowchartApproval/flowchartApproval/validate
Unica Optimize	<ul style="list-style-type: none"> ◦ WebSEAL junction/Campaign/optimize/ext_runOptimizeSession.do ◦ WebSEAL junction/Campaign/optimize/ext_optimizeSessionProgress.do ◦ WebSEAL junction/Campaign/optimize/ext_doLogout.do
IBM SPSS Modeler Advantage Enterprise Marketing Management Edition	WebSEAL junction/unica/rest/spssUser
Unica Platform data filters	WebSEAL junction/unica/servlet/DataFiltering.
Unica notifications	<ul style="list-style-type: none"> ◦ WebSEAL junction/unica/servlet/DataFiltering ◦ WebSEAL junction/unica/servlet/alertAJAXProxy ◦ WebSEAL junction/unica/notification/alertsCount
Unica Scheduler	WebSEAL junction/unica/servlet/SchedulerAPIServlet
Enable a logout of IBM Security Access Manager when a user logs out of an Unica application	<ul style="list-style-type: none"> ◦ WebSEAL junction/unica/j_spring_security_logout ◦ WebSEAL junction/unica/jsp/frameworklogout.jsp
Unica Platform	<ul style="list-style-type: none"> ◦ WebSEAL junction/unica/css/access_control.css ◦ WebSEAL junction/unica/api/manager/rest/v2/

Product or feature	Objects
	<ul style="list-style-type: none"> ◦ <code>WebSEAL junction/unica/api/manager/rest/v2/users</code> ◦ <code>WebSEAL junction/unica/api/manager/rest/v2/roles</code> ◦ <code>WebSEAL junction/unica/api/manager/rest/v2/utilities</code>
Asset-viewer	<code>/WebSEAL/<instance name>/<junction name>/asset-viewer/api/AssetPicker/object-mapping</code>

Configuration process roadmap: integrating Unica with a web access control system

Use this configuration process roadmap to scan the tasks required to integrate Unica with a web access control system. The Topic column provides links to the topics that describe the tasks in detail.

Table 48. Configuration process roadmap: integrating Unica with a web access control system

Topic	Information
Performing LDAP integration (on page 160)	Follow instructions for LDAP integration, stopping at the "Test synchronization" step.
Setting web access control connection properties in Unica (on page 160)	Set web access control integration properties on the Configuration page.
Restarting the web application server (on page 137)	This step is required to ensure that all of your changes are applied.
Testing web access control synchronization and Unica login (on page 161)	Verify that users and groups synchronize correctly in your web access control system and that you can log in to Unica.

Performing LDAP integration

Perform all of the steps required for LDAP integration.


Setting web access control connection properties in Unica

To configure web access control integration, you set some configuration properties.

On the **Settings & Configuration** page, set values of the properties as described in the following table.

See the related reference for details on how to set each property.

Table 49. Properties for configuring web access control integration

Property	Value
Unica Unica Platform Security Login method details	Select <code>web access control</code> .
Unica Unica Platform Security Login method details Web access control Additional header variables	<p>Specified comma separated variables are looked up in HTTP header, while logging through Web access control software. If audit log is enabled, these variables are captured and stored in the Authentication event under Audit logs. The captured HTTP variables can be seen by clicking "More" under "Event Details".</p> <p> Note: This property is available from version 12.1.0.3 onwards.</p>
Unica Unica Platform Security Login method details Web access control Username pattern	A Java™ regular expression used to extract the user login from the HTTP header variable in web access control software. You must XML-escape any XML characters in the regular expression. The recommended value for SiteMinder and IBM Security Access Manager is <code>\w*</code>
Unica Unica 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 IBM Security Access Manager uses <code>iv-user</code> . For IBM Security Access Manager, set this value to the user name component of the IBM® Raw string, not the IBM® HTTP string.
Unica General Navigation Unica Platform URL	<p>Set to <code>http://sm_host:sm_port/sm_realm/unica</code></p> <p>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.

Testing web access control synchronization and Unica login

Follow this procedure to test your integration.

1. Log in to your web access control system with an LDAP account that has been synchronized into your web access control system and has access to Unica Platform.
2. Verify that:

- Users are imported as expected
 - Groups are imported as expected
 - Unica group memberships match the expected mapping to LDAP groups
3. Point your browser to the Unica Platform URL and log in.

You should be able to access Unica without being presented with the Unica login screen.

4. Use the following guidelines to resolve problems when your web access control software is Netegrity SiteMinder.
- If you see an Unica login screen, the user account with which you logged in might not have been synchronized into SiteMinder.
 - If you are not able to access Unica, 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 Unica URLs in SiteMinder.
 - If you can access Unica, 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 Unica Platform use the same path to refer to Unica Platform.

Verify additional headers in audit logs

Ensure that audit logs are enabled. Under "Web access control| Additional header variables" property, specify the HTTP header variables names to capture. After a successful login, verify the Audit event reports and check the Event Details for captured variables.

Configuring integration with an SSL type of WebSEAL junction

Follow this procedure to configure Unica Platform integration with IBM Security Access Manager using an SSL type of WebSEAL junction.

For details on these procedures, see the documentation provided with IBM Security Access Manager and your web application server.

1. Generate or purchase SSL certificates and configure your web application server to use them.
2. Create a webSEAL certificate and configure IBM Security Access Manager to use it.
3. Import your webSEAL certificate into your web application server.
4. Import your web application server certificate into IBM Security Access Manager.
5. Create an SSL type of WebSEAL junction in IBM Security Access Manager.

If you install multiple Unica products, create a separate junction for each product.

6. Set the navigation URL configuration property on the **Settings & Configuration** page for each installed product.

The value should reflect the webSEAL junction used for that product. Follow this pattern:

```
https://machine_name_or_IP_address.domain_name:port_number/webSEAL_junction/
context-root
```

To access Unica, use a URL such as the following:

```
https://machine_name_or_IP_address.domain_name:port_number/webSEAL_junction//
unica
```

7. Unprotect URLs in IBM Security Access Manager as described elsewhere in this guide.

Alert and notification management

Unica Platform provides support for system alerts and user notifications sent by Unica 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 Unica Platform has been configured to send them.

Within Unica Platform, the Unica Scheduler uses the notification feature.

Alert and notification subscriptions

Users can choose to have system alerts and notifications delivered in emails, if Unica 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 Unica. Users cannot control these by changing their subscriptions.

When users log in to Unica, the **System alerts** window is displayed only if there are any new or unread alerts. Users can mark an alert as read by selecting the alert and clicking **Mark as read** in the **System alerts** window.

Setting system alert and notification subscriptions

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

1. Log in to Unica 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 notifications in Unica

Follow this procedure to configure the Unica Platform to send system alert and notification emails to users. You must have an email server set up before you start.

Obtain the following information about your mail server.

- The protocol used by your mail server.
- The port on which the mail server listens.
- The name of the machine that hosts your mail server.
- Whether your mail server requires authentication.
- If your mail server requires authentication, an account name and password on the mail server.



Tip: See the related references if you need additional details about performing this procedure.

1. If your mail server requires authentication, save a mail server account name and password as a data source in a Unica Platform user account.

Use an internal Unica Platform user account, not a user imported from an LDAP server.

Make a note of the Unica Platform user name and the data source name, as you will use them in step 3.

2. Log in to Unica as a user with administrative privileges in Unica 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.

Implementation of one-way SSL

This section describes one-way SSL in Unica.

Any communication that needs to be secured between two applications connecting over a network can be transmitted using the Secure Sockets Layer (SSL) protocol.

SSL provides secure connections by:

- Allowing an application to authenticate the identity of another application
- Using a private key to encrypt and decrypt data transferred over the SSL connection

When applications are configured for SSL, web traffic is over HTTPS instead of HTTP as reflected in the URLs.

When processes communicate with each other, the process making a request acts as the client and the process responding to a request acts as the server. For complete security, SSL should be implemented for all forms of communication with Unica 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.

Overview of SSL certificates

Read this section to understand SSL certificates in general.

What is a certificate?

A certificate is a digital signature that identifies the server as some named entity. Certificates can be signed by a certificate authority (CA) that vouches for the identity of the server, or they can be self-signed. Verisign or Thawte are examples of CAs. A self-signed certificate is one where the CA is the same entity that the certificate claims to identify.

Server-side certificates

Every server that is intended to provide SSL communication, whether it is an application server or an Unica application such as the Unica 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 a HCL 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 may issue a warning if these two names do not match.

When a browser accesses a HCL 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 Unica

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

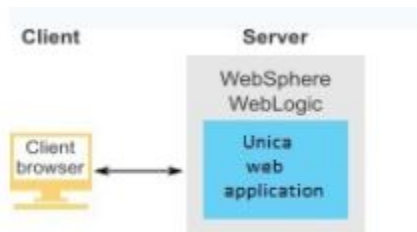
Most Unica applications consist of two parts.

- The web application. The web application is the component that users access through a browser.
- The server (for example, the Unica Campaign listener and the Unica Platform API server). This component is accessed programmatically.

The following examples and diagrams illustrate the roles played by HCL components in various communications.

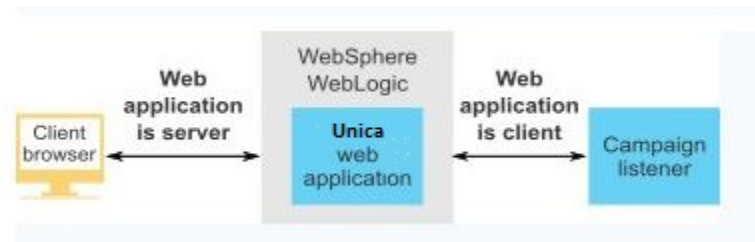
Example 1 - Communication between a browser and an Unica web application

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



Example 2 - Communication between components of one Unica application

The two components of a single Unica application can also communicate with each other programmatically. For example, when the Unica Campaign web application sends a request to the Unica Campaign listener, the Unica Campaign web application is the client and the listener is the server.



Example 3 - Unica components playing both roles

An Unica 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 Unica

Many application components can act as both server and client during normal operations, and some 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.

applications do not require a truststore when they act as a client making one-way SSL requests to an server component.

Java™ component acting as a server

For applications written in Java™, using the JSSE SSL implementation, and deployed on an application server, you must configure the application server to use your certificate. The certificate must be stored in JKS format.

You cannot use the default certificate provided with the application server.

You can create JKS certificates for your Java applications using Java keytool.

C++ component acting as a server

The Campaign listener, Optimize server component are written in C++, and require a certificate generated by OpenSSL.

Java™ component acting as a client

For applications written in Java™ and deployed on an application server, no truststore is needed. For ease of configuration, 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 required. The Campaign listener fall into this category.

How many certificates?

Ideally, you should use a different certificate for every machine that hosts an component acting as a server.

If you do not want to use multiple certificates, you can use the same certificate for all the components acting as servers. If you use one certificate for all applications, when users access applications for the first time, the browser asks whether they want to accept the certificate.

Configuration process roadmap: implementing SSL in Unica

Use this configuration process roadmap to scan the tasks required to implement SSL in Unica. The Topic column provides links to the topics that describe the tasks in detail.

Table 50. Configuration process roadmap: implementing SSL in Unica

Topic	Information
Obtain certificates (on page 173)	Obtain or create certificates if you prefer not to use the default certificates provided by HCL and your application server.
Configure your web application servers for SSL (on page 174)	Enable an SSL port in every application server where a HCL application is deployed. If you are not using the application server default certificate, configure it to use your certificate.
Configure HCL Unica for SSL (on page 175)	Set configuration properties in Unica.
Verifying your SSL configuration (on page 184)	Log in to each of your Unica applications.

Certificates for SSL

This procedure describes how to create and configure your own certificates. Perform a similar procedure for every Unica you configure to use SSL. If you are configuring the Unica Campaign + Engage integration, see the Unica Campaign and Engage Integration Guide for IBM Marketing Cloud.

You can obtain or create certificates in several ways. You can create self-signed certificates or you can obtain certificates from a certificate authority (CA).

Self-signed certificates

You can create self-signed certificates.

For C++ components acting as a server, use openssl to create a `.pem` certificate. The Campaign listener implements SSL using the HCL openssl library. The openssl is installed with Campaign and it includes a command-line program called `openssl` that can create a certificate file.

For Java components acting as a server, use Java keytool to create a JKS certificate.

Certificates from the certificate authority

You can obtain certificates from a certificate authority (CA).

You can use the openssl to create requests that you can then send to a CA to create signed certificates. Or, you can obtain signed certificates entirely provided by the CA.

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

Obtain or create certificates

Complete the following procedures to create self-signed certificate files and use with HCL Unica.

1. Create a certificate for a C++ application HCL Unica components.
2. Create a certificate for a C++ application Java Unica components.

Create a certificate for a C++ application HCL Unica components

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 details on using this program, see the OpenSSL documentation. You can also access the help by entering `-help`, when you run the program.

Complete the following steps to create a self-signed certificate and configure a C++ HCL Unica component for SSL.

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. Here is a sample command that creates a key named `key.pem`.

```
set OPENSLL_CONF=CAMPAIGN_HOME\bin\openssl.cnf
```

```
openssl genrsa -out key.pem 4096
```

3. Generate a request. Here is a sample command that creates a key named `request.pem`.

```
openssl req -config openssl.cnf -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:

```
openssl req -x509 -key key.pem -in request.pem -days 1000 -out certificate.pem
```

4. Generate a certificate. The following sample 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.

```
openssl 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:

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

5. Create new certificate file example campaign.pem.
6. Copy key.pem and certificate.pem content into this file separated by new line.

Create a certificate for Java HCL Unica components

HCL Unica web application components written in Java use the JSSE library. The Sun JDK includes a program called `keytool`, which can create a certificate file. For details on using this program, see the Java documentation. You can also access the help by entering `-help` when you run the program.

Complete the following steps to create a self-signed certificate and configure a Java HCL Unica component for SSL.

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 sample 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 require it when you configure the application server.
 - Make a note of the `-alias` value (`UnicaClientIdentity` in the example) as you require it for the rest of this procedure.
 - The common name (CN) in the distinguished name must be the same as the host name used to access HCL Unica. For example, if the URL for HCL Unica is `https://hostName.companyDomain.com:7002/unica/jsp`, then the CN must 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 sample command creates a certificate named `UnicaCertificate.cer`. The value of `-alias` is the alias that you set for the identity keystore (`UnicaClientIdentity` in the example).

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

4. Generate a trusted keystore based on the certificate you created. The following sample 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 require it when you configure the application server.

Import Open SSL certificate into java key store

```
keytool -import -alias ListenerKey -file CAMPAIGN_HOME\bin\certificate.pem -keystore PlatformClientIdentity.jks
-storepass password
```

```
keytool -import -file CAMPAIGN_HOME\bin\certificate.pem -alias ListenerKey -keystore <APP_SERVER_JAVA>\jre\lib
\security\cacerts
```

How to obtain signed certificates

You can either use the OpenSSL and keytool programs to create requests to send to a CA to create signed certificates or you can obtain signed certificates entirely provided by the CA.



Note:

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

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

Creating and configuring certificates for a clustered environment

This procedure describes how to create and configure your own certificates for a clustered environment.

The Campaign web application must be configured for SSL by using default certificates.

The following procedure describes how to create and configure self-signed certificates for Unica Campaign and Unica Platform.

In a clustered environment where there is an IBM HTTP Server in front of the Unica Campaign web application and Campaign listener, follow these steps to configure the Campaign listener in SSL.

You can use these steps as a guide for configuring certificates for other Unica products.

This procedure is applicable for the default certificates that are provided by the IBM WebSphere Application Server. If you are using custom security certificates, you must follow the steps for the custom certificates used by the IBM WebSphere Application Server.

To configure the IBM HTTP Server in SSL, complete the following steps.

1. Use GSKit to generate SSL certificates as follows.

a. Create and initialize a new key database.

For example:

```
gsk8capicmd_64 -keydb -create -populate -db IHS.kdb -pw password -stash
```

The `-stash` option is required for Unica Campaign.

b. Use GSKit to generate a self-signed certificate for Unica Campaign and store it in the key database, as follows.

For example:

```
gsk8capicmd_64 -cert -create -db IHS.kdb -dn "CN=*.in.ibm.com" -expire 3650 -pw password -size 1024 -label key -default_cert yes
```

c. Extract the public part of the certificate to a file.

For the clients to trust a certificate, its public part needs to be distributed to the clients and stored in their key databases. In this step, you export the public part of the Unica Campaign certificate. You import it in a later step.

For example:

```
gsk8capicmd_64 -cert -extract -db IHS.kdb -stashed -label key -target IHS.arm
```

d. Enable the following module in the `httpd.conf` file.

For example:

```
LoadModule ibm_ssl_module modules/mod_ibm_ssl.so

Listen 443

<VirtualHost *:443>
    SSLEnable
</VirtualHost>

KeyFile /data/webrowsers/IBM/IHS/ssl/IHS.kdb

SSLStashFile /data/webrowsers/IBM/IHS/ssl/IHS.sth

SSLDisable
```

e. Provide the key file path in the `httpd.conf` file.

f. Restart the IBM HTTP server.

2. Generate keystore database files for the server that hosts the Unica Campaign listener.

- a. On the server that hosts the Unica Campaign listener, run the following commands from any location and note the path.

```
gsk8capicmd_64 -keydb -create -populate -db Key.kdb -pw password -stash
gsk8capicmd_64 -cert -create -db Key.kdb -dn "CN=*.in.ibm.com" -expire 3650 -pw password -size
1024
-label key -default_cert yes
gsk8capicmd_64 -cert -extract -db Key.kdb -stashed -label key -target Key.arm
```

- b. Verify that the following files are generated in the location from where you ran the above commands.

- `Key.arm`
- `Key.crl`
- `Key.kdb`
- `Key.rdb`
- `Key.sth`

3. Import the `Key.arm` and `HIS.arm` files into the application server where the Campaign web application is deployed.

- a. Copy the `Key.arm` and `HIS.arm` files to the Campaign web application server.
- b. Add the `Key.arm` and `HIS.arm` files in the **NodeDefaultTrustStore** of the WebSphere Application Server by completing the following steps:
- i. Click **Security > SSL Certificate and key management > Key stores and certificates**.
 - ii. Click **NodeDefaultTrustStore > Signer certificates**.
 - iii. Click **Add** and provide the **Alias** and the path where the `Key.arm` and `HIS.arm` files are copied.
 - iv. Click **OK**.

4. Extract the Personal and Signer certificates for the IBM WebSphere Application Server

- a. Click **Security > SSL Certificate and key management > Key stores and certificates**.
- b. Click **NodeDefaultTrustStore > Personal Certificates**.
- c. Select the default certificate.
- d. Add the Personal Certificate file name along with the valid path in the Unica Campaign web application server. For example, `/opt/HCL/HCLUnica101/ClientPersonal.cer`.
- e. Click **OK**.
- f. Click **NodeDefaultTrustStore > Signer Certificates**.
- g. Select the default certificate.
- h. Add the Signer Certificate file name along with the valid path in the Unica Campaign web application server. For example, `/opt/HCL/HCLUnica101/ClientSigner.cer`.
- i. Navigate to the folder and verify the both certificates are present in the folder.

5. Import the Personal and Signer certificates into the Unica Campaign listener and HCL HTTP Server keystore databases.

a. Copy the `ClientPersonal.cer` and `ClientSigner.cer` certificates to the listener server. You can use the same location where the `key.kdb` file was created.

b. Import the Personal and Signer certificates to the listener keystore database by using the `gsk8capicmd_64` command from the location where the listener keystore database (`key.kdb`) was created.

```
gsk8capicmd_64 -cert -add -db Key.kdb -stashed -label ClientPersonalKey -file
ClientPersonal.cer
gsk8capicmd_64 -cert -add -db Key.kdb -stashed -label ClientSignerKey -file ClientSigner.cer
```

c. Copy the `ClientPersonal.cer` and `ClientSigner.cer` certificates to the HCL HTTP Server. You can use the same location where the `IHS.kdb` file was created.

d. Import the Personal and Signer certificates to the listener keystore database by using the `gsk8capicmd_64` command from the location where the HCL HTTP Server keystore database (`IHS.kdb`) was created.

6. Import the Campaign listener key in the HCL HTTP Server keystore database and import the HCL HTTP Server key in the Campaign keystore database.

a. Copy the HCL HTTP Server key (`IHS.arm`) to the listener server.

b. Import the HCL HTTP Server key to the listener keystore database by using the `gsk8capicmd_64` command from the location where the Campaign listener keystore database (`key.kdb`) was created.

```
gsk8capicmd_64 -cert -add -db Key.kdb -stashed -label IHSKey -file IHS.arm
```

c. Copy the Campaign listener key (`key.arm`) to the listener server.

d. Import the Campaign listener key to the HCL HTTP Server keystore database by using the `gsk8capicmd_64` command from the location where the HCL HTTP Server keystore database (`IHS.kdb`) was created.

```
gsk8capicmd_64 -cert -add -db IHS.kdb -stashed -label IHSKey -file Key.arm
```

7. Restart the HCL Campaign application server, the HCL HTTP server and then start the Unica Campaign Listener.

Configure your web application servers for SSL

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

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

Ensuring cookie security


Some cookies may not be properly secured in the client browser. Not securing cookies leaves the application vulnerable to man-in-the-middle and session hijacking attacks. To fix this issue, take the following precautions.

- Enforce the use of SSL at all times to reduce the risk of cookies being intercepted on the wire.
- In the web application server, set the `secure` and `httponly` flags on all cookies.
 - The `secure` flag tells the browser to send the cookie only over an HTTPS connection. You must enable SSL on all applications that communicate with each other if you set this flag.
 - The `httponly` flag prevents cookies from being accessed through a client side script.

Setting the flags for SSL in WebSphere®

To set the `secure` and `httponly` flags in WebSphere®, use the following procedure.


You set the `secure` and `httponly` flags in the WebSphere® administrative console.

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

1. At the application level for Unica Platform, navigate to **Session Management** and click **Enable cookies**.
2. Check **Restrict cookies to HTTPS sessions** and **Set session cookies to HTTPOnly to help prevent cross-site scripting attacks**.
3. Save and apply your changes.
4. Stop and re-start the Unica Platform application.

Setting the flags for SSL in WebLogic

To set the `secure` and `httponly` flags, use the following procedure.

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

1. If Unica Platform is deployed and running, stop and undeploy it.
2. Extract the Unica Platform WAR file.
3. Edit the `weblogic.xml` file to set the `secure` and `httponly` flags.
4. Recreate the Unica Platform WAR file, redeploy, and re-start.

Configure HCL Unica for SSL

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

When you access your Unica 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 Unica Platform

Follow this procedure to configure SSL in Unica Platform.

1. Log in to Unica and click **Settings > Configuration**.
2. Set the value of the `General | Navigation | Unica Platform URL` property to Unica Platform URL.

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

where:

- `host` is the name or IP address of the machine on which Unica Platform is installed
- `domain` is your company domain in which your Unica products are installed
- `SSL_Port` is the SSL port in the application server on which Unica Platform is deployed

Note `https` in the URL.

3. Locate the properties under the `Navigation` category for each of your installed Unica 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 Unica Platform with LDAP integration."
5. If you plan to use the data filtering feature, perform the procedure described in "Configuring SSL in Unica Platform with data filters."

Configuring SSL in Platform for a clustered environment

Follow this procedure to configure SSL in Platform in a clustered environment.

1. Log in to HCL Unica and click **Settings > Configuration**.
2. Under `Affinium | Manager | Navigation`, set **Unica Platform URL** to the Unica Platform URL.

For example: `https://<IHS_Host>/unica`.

3. Under `Affinium | Campaign | Navigation`, set **serverURL** to the Unica Campaign URL.

For example: `https://<IHS_Host>/Campaign`.

4. Under `Affinium | Campaign | server`, set **fullContextPath** to the Unica Campaign URL.

For example: `https://<IHS_Host>/Campaign`.

5. Under `Affinium | Campaign | unicaACLlistener`, set **serverhost** to the `<IHS Host>` and set **useSSL** to `True`.

Configuring SSL in Unica Platform with LDAP integration

Follow this procedure to configure SSL in Unica Platform.

1. Perform the procedure described in "Configuring SSL in Unica Platform" if you have not done so already.
2. Log in to Unica and click **Settings > Configuration**.

The Configuration page appears.

3. Navigate to the Unica | Unica Platform | Security | Login Method details | LDAP category and set the value of the `Require SSL for LDAP connection` property to `true`.

This setting requires Unica Platform to connect to the LDAP server using SSL when users log in.

4. Navigate to the Unica | Unica 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 Unica Platform to connect to the LDAP server using SSL when it synchronizes with the LDAP server.

Configuring SSL in Unica Platform with data filters

When Unica Platform is deployed with SSL and you plan to use the data filtering feature, you must perform this procedure to add the SSL options that perform hand shaking.

1. Perform the procedure described in "Configuring SSL in Unica Platform" if you have not done so already.
2. Obtain the following.
 - A copy of the certificate file you created in Obtaining or creating certificates (on page)
 - 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 Unica Platform installation.

The `setenv` script specifies the Java™ instance used by Unica 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 Unica Plan

Follow this procedure to configure SSL in Unica Plan.

1. Log in to Unica and click **Settings > Configuration** .
2. Set the value of the `Marketing Operations | navigation | serverURL` property to the URL of the Unica Plan web application.

For example: `serverURL=https://host:SSL_port/plan`

where:

- `host` is the name or IP address of the machine on which Unica Plan is installed.
- `SSL_Port` is the SSL port of the Unica Plan 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 Unica Plan 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 Unica Plan is installed.
- `SSL_Port` is the SSL port of the Unica Plan web application

Note the `https` in the URL.

5. To enable Adobe™ Acrobat Online Markup functionality to work with Unica Plan 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 Unica Plan is installed
- `SSL_Port` is the SSL port of the Unica Plan web application

Note the `https` in the URL.

6. Save and close the `plan_config.xml` file.

Configuring SSL in Unica Campaign

Follow this procedure to configure SSL in Unica Campaign.



Note: If you are configuring SSL in Unica Campaign, you must also configure the Campaign Listener in SSL. If you do not set the Campaign Listener in SSL the schedule flowchart status might be shown as `Unknown`.

1. Open the `config.xml` file in a text or XML editor.

The `config.xml` file is in the `conf` directory under your Unica Campaign installation.

2. Set the following values in the `config.xml` file.

```
◦ unicaServerSSLFile = PATH_TO_OPENSSL_PEM/campaign.pem
```

3. Save and close the `config.xml` file.

4. Log in to Unica Platform and click **Settings > Configuration**.

The Configuration page appears.

5. Set the value of the `Campaign | unicaACLListener | 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 cipher list in Unica Campaign

Prerequisite: Unica Campaign must be configured with SSL.

If Unica Campaign application and Listener are configured with SSL options as `TRUE`, then by default 98 ciphers are supported to enable the SSL communication between Unica Campaign application(Server) and listener.

To disallow weak ciphers from this default cipher list, users can use `<SSLCipherList>` tag or property in `config.xml` file.

To remove support of weak ciphers, users must add the following line in `config.xml` file. It specifies that support to default ciphers excludes `AES256-SHA, CAMELLIA256-SHA, AES128-SHA, SEED-SHA, CAMELLIA128-SHA, DES-CBC3-SHA, IDEA-CBC-SHA`.

```
<property name="SSLCipherList"><value>DEFAULT:!AES256-SHA:!CAMELLIA256-SHA:!AES128-SHA:!SEED-SHA:!CAMELLIA128-SHA:!DES-CBC3-SHA:!IDEA-CBC-SHA</value></property>
```

This disables the above-mentioned ciphers, which are included in `<SSLCipherList>` tag of `config.xml` file.

If clients or users do not mention the `SSLCipherList` tag in `config.xml` file, then the default cipher list is considered and 98 ciphers are supported.



Note: The listener will not start and the following errors are generated in `unica_aclsnr.log` file, if users or clients disable any cipher which is required by certificate or browser.

```
Error enabling SSL connection.
```

```
SOCKET BIND port=4664: ERRNO 10048: Unknown error
```

Configuring SSL in Unica Campaign for a clustered environment

Follow this procedure to configure SSL in Unica Campaign listener server in a clustered environment.

1. Open the `config.xml` file for the listener server in a text or XML editor.

The `config.xml` file is in the `conf` directory under your Unica Campaign installation.

2. Set the following values in the `config.xml` file.
 - Set `configurationServerBaseURL` to the Campaign SSL URL. This is the HCL HTTP Server URL.
 - Set `unicaServerSSLFile` to the path where the Password file is saved.
 - Set `unicaServerSSLFilePwd` to the path where the Password file is saved.

For example:

```
<configuration name="bootstrap">
  <category name="bootstrap">
    <property name="suiteName"><value>Affinium</value></property>
    <property name="clientType"><value>HTTP</value></property>
    <!-- configurationServerBaseURL value will be set by AffiniumSuite assembly installer -->
    <property
name="configurationServerBaseURL"><value>https://<IHS_Host>/Campaign</value></property>
    <property name="trustedApplication"><value>>false</value></property>
```

```

<property name="unicaClientKeystore"><value></value></property>
<property name="unicaClientKeystorePwd"><value></value></property>
<property name="unicaServerSSLFile"><value>/PATH_TO_OPENSSL_PEM/campaign.pem</value></property>
<property name="unicaServerSSLFilePwd"><value></value></property>

</category>
</configuration>

```

3. Save and close the `config.xml` file.

Configuring Campaign in SSL and Campaign Listener in non-SSL

If your setup has Campaign in SSL and the Campaign Listener in the non-SSL mode, you must configure settings for the applications to work seamlessly.

The Campaign web application must be configured in SSL by using the default certificates.

All configurations are applicable to the WebSphere Application Server for Campaign. Multiple steps are involved to configure the SSL and non-SSL setup. Each step might have more substeps to be completed.

To configure Campaign in SSL and Campaign Listener in non-SSL, complete the following steps:

Complete the following steps.

Table 51. Configuring Campaign in SSL and Campaign Listener in non-SSL

#	Step	Substeps
1	Generate and use <code>.pem</code> (certificate) file.	<p>Run the following commands from and location and note the paths. Create new certificate file example <code>campaign.pem</code> (copy <code>key.pem</code> and <code>certificate.pem</code> content into this file separated by new line)</p> <pre> set OPENSSL_CONF=CAMPAIGN_HOME\bin\openssl.cnf openssl genrsa -out key.pem 4096 openssl req -config openssl.cnf -new -key key.pem -out request.pem openssl req -config openssl.cnf -x509 -key key.pem -in request.pem -days 1000 -out certificate.pem </pre> <p>The following files are generated at the location from where you ran the commands.</p> <ul style="list-style-type: none"> • <code>key.pem</code> • <code>request.pem</code> • <code>certificate.pem</code> • <code>campaign.pem</code>
2	Import the <code>campaign.pem</code> file into the application server where	<ol style="list-style-type: none"> a. Copy the <code>campaign.pem</code> file to the Campaign web application server. b. Add the <code>campaign.pem</code> file in the NodeDefaultTrustStore of the WebSphere Application Server by completing the following steps:

#	Step	Substeps
	the Campaign web application is deployed.	<p>i. Click Security > SSL Certificate and key management > Key stores and certificates.</p> <p>ii. Click NodeDefaultTrustStore > Signer certificates.</p> <p>iii. Click Add and provide the Alias and the path where the <code>campaign.pem</code> file is copied.</p> <p>iv. Click OK.</p> <p>The listener key is added to the application server.</p>
3	Modify <code>config.xml</code> file on the listener server.	<p>Provide the following information:</p> <ul style="list-style-type: none"> • configurationServerBaseURL: Provide the Campaign SSL URL. • unicaServerSSLFile: Provide the <code>PATH_TO_OPENSLL_PEM/campaign.pem</code> file path. • unicaServerSSLFilePwd: Provide the corresponding <code>password</code> file path. <pre data-bbox="527 867 1425 1766"> <configuration name="bootstrap"> <category name="bootstrap"> <property name="suiteName"><value>Affinium</value></property> <property name="clientType"><value>HTTP</value></property> <!-- configurationServerBaseURL value will be set by AffiniumSuite assembly installer --> <property name="configurationServerBaseURL"> <value>https://eagle191.hcl.com:9447/Campaign</value> </property> </category> <property name="trustedApplication"><value>>false</value></property> <property name="unicaClientKeystore"><value></value></property> <property name="unicaClientKeystorePwd"><value></value></property> <property name="unicaServerSSLFile"> <value>PATH_TO_OPENSLL_PEM/campaign.pem</value> </property> <property name="unicaServerSSLFilePwd"> <value> password </value> </property> </category> </configuration> </pre>

#	Step	Substeps
4	In the unicaACLis- tener settings set useSSL to TRUE .	-
5	Restart the Campaign Application Server and the Campaign Listener.	-

Configuring SSL in Unica Optimize

Follow this procedure to configure SSL in Unica Optimize.

1. Open the `config.xml` file found in the `conf` directory of your Unica Optimize installation directory in a text or XML editor.
2. Set the value of `unicaServerSSLFile` to the full path of the certificate you are using.
3. Save and close the `config.xml` file.
4. Set the value of the `Campaign | unicaACOListener | useSSL` configuration property to `yes`.
5. If you are using the Unica Optimize command-line tool `ACOOptAdmin`, perform the following steps.
 - a. Obtain the following.
 - A copy of the certificate file you created in Obtaining or creating certificates (*on page*)
 - The certificate password
 - b. Place the certificate file in the `JAVA_HOME/jre/lib/security` directory, where `JAVA_HOME` is the Java™ directory specified in the `ACOOptAdmin` script.
 - c. Use the `keytool` program to import the certificate into the `cacerts` file for your Java™ instance.

You can use the following example command as a guide.

```
keytool -import -trustcacerts -file name_of_your_certificate.cer -keystore cacerts
```

Enter the certificate password when prompted.

Configuring SSL in Unica Interact

You can configure SSL communication for Unica Interact in three areas, although there is a significant performance cost if you do this.

The areas in that can use SSL are as follows.

- Design environment as the client and Runtime environment as the server.

Use https in the URL referencing the Unica 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 Unica Platform as the server.
- Your touchpoint as the client and the Runtime environment as the server.

Specify the HTTPS URL with the `getInstance` method. If using a load balancer, you might need to configure your load balancer for SSL as well.

- If the Unica Interact design server and Runtime server are on separate hosts using SSL, import the security certificates on the two servers to enable the SSL handshake to take place.



Important: There is a performance cost if you configure any part of Unica Interact to communicate using SSL. It is not recommended to configure Unica Interact to use SSL.

Configuring SSL in Unica Collaborate

After Unica Campaign is configured to use SSL, no additional configuration is required to configure Unica Collaborate for SSL.

Configuring SSL in Reports

Follow this procedure to configure SSL in Reports.

1. Configure Cognos® with SSL as described in the Cognos® documentation.
2. Configure Apache with SSL as described in the Apache documentation.
3. Register the Cognos® certificate with Unica as described in the Cognos® documentation.
4. Register the Unica certificates with Cognos® as described in the Cognos® documentation.

Configuring SSL in Digital Analytics for On Premises

Digital Analytics for On Premises does not accept any requests: it always acts as the client in HTTP and HTTPS communications to resolve page titles on the web site being analyzed. If you need to resolve page titles for a site that uses SSL, you only need to ensure that the URL entered in the profile options for the website or clustered servers being analyzed is correct and that the URL includes the HTTPS protocol.

SDigital Analytics for On Premises does not communicate with Unica Platform.

Verifying your SSL configuration

Follow this procedure to verify your SSL configuration.

1. Start each of your Unica applications.
2. Log in to Unica and access each of your installed Unica web applications.

3. For Unica 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 Unica server components and verify that the certificate information you receive is as expected.

For example, if the Unica 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 Unica.

- OpenSSL Documentation - <https://www.openssl.org/>
- Java keytool documentation - <http://docs.oracle.com/javase/7/docs/technotes/tools/solaris/keytool.html>
- List of certificate authorities - https://curlie.org/Computers/Security/Public_Key_Infrastructure/PKIX/Tools_and_Services/Third_Party_Certificate_Authorities/

Quality of protection (QoP) settings for WebLogic

You must set the QoP settings when you configure HCL Unica applications to use SSL.

The following QoP settings are supported for WebLogic:

- TLS11
- TLS12

To change the QoP settings, complete the following steps:

Append the following option to the `JAVA_OPTIONS` variable:

- For TLS11- `-Dweblogic.security.SSL.protocolVersion=TLSv1.1`
- For TLS12- `-Dweblogic.security.SSL.protocolVersion=TLSv1.2`

Quality of protection (QoP) settings for WebSphere

You must set the QoP settings when you configure HCL Unica applications to use SSL.

The following QoP settings are supported for WebSphere:

- SSL_TLS
- SSL
- TLS
- TLSv1
- SSL_TLSv2

- TLSv1.1
- TLSv1.2

To change the QoP settings, complete the following steps:

1. Go to **Security > SSL Certificate and key management > SSL configurations**
2. Select the required SSL configuration.
3. Under **Additional Properties**, click **Quality of protection (QoP) settings**.
4. In the **Quality of protection (QoP) settings** pane, select the required QoP settings from the drop-down list for **Protocol**.
5. Click **Save**.
6. In the `ssl.client.props` file located in the `WAS_install\profiles\AppSrv01\properties` folder, update the following:

```
com.ibm.ssl.protocol=<Specify required QoP settings>
```

7. Restart the application server.

Security framework for Unica APIs

Unica Platform provides the security framework for the APIs implemented by Unica products.

A set of configuration properties on the **Settings > Configuration** page enables developers to set the following security for the APIs provided by Unica products.

- For a specific product API, you can block access to the product.
- For a specific product API, you can require HTTPS for communication between the specified API and the product.
- For a specific product API, you can require authentication for communication between the specified API and the product.

The configuration properties that control API security are located under the **Unica Platform | Security | API management** category. Each product has a configuration property template that you can use to create new security settings for the APIs provided by that product.

You can set and change the security settings for an API as appropriate for unit testing or deployment or during the overall lifecycle of APIs.

The security framework currently supports APIs for Unica Campaign only.

The Unica Platform security framework supports the following two authentication options for accessing protected APIs. You can use either one, depending on your environment.

- Internal users who are registered with Unica Platform can be authenticated using their Unica Platform login credentials to obtain a secure token.
- External users who are part of a federation that Unica Platform is set up to use can be authenticated through the Identity Provider server.

Internal user authentication with the Unica Platform login API

To authenticate internal users in client applications, use the Unica Platform `login` API to generate secure tokens. You can then invoke any protected APIs by passing the required parameters in the request header, in addition to the parameters expected by the API itself.

The security filter intercepts these protected requests, validates them, and then passes them through for processing.

After the Unica Platform user is authenticated, the Unica Platform security filter adds the user's login name to the request as an attribute of the `USER_NAME_STRING` key before passing it to the product for processing.

The secure tokens have a default life span of 15 seconds. After the life span of the token expires, it cannot be used to invoke a protected API. Each time the Unica Platform `login` API is invoked for a user, all previous security tokens for that user are invalidated.

You can change the life span of secure tokens by setting the value of the **Token lifetime** property located on the **Settings > Configuration** page under the **General | Miscellaneous** category.

Example URL

```
http[s]://host:port/unica/api/manager/authentication/login/
```

Header parameters

Table 52. Header parameters for the login API with internal users

Parameter	Description
<code>m_user_name</code>	The internal user's Unica Platform login name.
<code>m_user_password</code>	The internal user's Unica Platform password in plain text.

Response

When login succeeds, the response is HTTP 200 with the following JSON data.

- `m_tokenId` - randomly generated token
- `m_user_name` - user name of the logged in user
- `createDate` - timestamp in the format that is shown in the following example, where the time zone is IST:

```
Mon Jul 06 18:23:35 IST 2015
```

When login fails with bad credentials, the response is HTTP 401 (unauthorized). When the `login` API is configured to be blocked, the response is 403 (forbidden). When the `login` API is configured to use HTTPS and if it is invoked on HTTP, the response is 403 (forbidden).

To log out internal users, use the Unica Platform `logout` API.

Internal user logout with the Unica Platform logout API

Use the Unica Platform `logout` API to log out internal users and delete the secure token.

The `logout` API is protected by default. The authentication parameters are expected in the request header against pre-defined keys.

Example URL

```
http[s]://host:port/unica/api/manager/authentication/logout/
```

Header parameters

Table 53. Header parameters for the logout API

Parameter	Description
<code>m_user_name</code>	The internal user's Unica Platform login name.
<code>m_tokenId</code>	The secure token obtained through authentication.
<code>api_auth_mode</code>	Use the value <code>manager</code> for internal users.

Response

When authentication succeeds, the response is HTTP 200, and the secure token is deleted. If the response is HTTP 200, the client application should confirm the logout.

When authentication fails, the response is HTTP 401.

External user authentication and logout through a federation

When Unica Platform is integrated with a supported federation, users can log in to their own system, and the client application gets a token through the Identity Provider (IdP) server provided by Unica Platform.

After a federated user is authenticated, their corresponding Unica Platform login name is added to the request as an attribute of the `USER_NAME_STRING` key.

Log out should be done at the IdP server.

Header parameters

The following table describes the header parameters to use when authenticating through the IdP server provided by Unica Platform.

Table 54. Header parameters with a federation

Parameter	Description
f_userId	User ID in the federation.
f_clientId	Client ID in the federation.
f_spId	Service provider ID in the federation.
f_tokenId	Single sign-on token from the IdP server.
api_auth_mode	Use the value <code>f_sso</code> for federated authentication.

Response

The response is `HTTP 200`, with additional items depending on the API.

Data filter creation and management

Data filters make it possible to restrict the customer data that an Unica user can view and work with in Unica applications. You can think of the data you secure with a data filter as a data set defined by the fields in your customer tables that you specify.

The various Unica applications use data filters in different ways. See the documentation for the individual products to determine whether the product uses data filtering, and if so, the details of how data filtering works within that product.

Overview of data filter creation

Unica Platform provides the following features that Unica 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 Unica users and groups are available for assignment to data filters.

You can assign multiple users and groups to a single data filter, and you can also assign a user or a group of users to multiple data filters.



Note: Groups do not acquire the data filter assignments of their subgroups.

A user who is assigned to multiple data filters sees all of the records allowed by all of the data filters.

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

Unica Platform provides a utility, `datafilteringScriptTool`, that processes XML to create the data filters in the Unica Platform system tables. Depending on how you write the XML, you can use this utility in two ways: automatic generation and manual specification.

Automatic generation

The `datafilteringScriptTool` utility can automatically generate data filters from a database table or view accessible using JDBC. The utility automatically creates data filters based on unique combinations of values in fields that you specify in the XML (one data filter for each unique combination).

You might want to use this method if you must create many data filters based on unique combinations of values in different fields.

Manual specification

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

You might want to use this method if you want to create a set of data filters that does not include every unique combination of field values.

Two ways to assign users and groups: in the user interface and in the XML

You have two options for assigning users and groups to data filters: through the user interface or in the XML you use to create the data filters. Assigning users in the XML is a useful method when you have many users, each of whom requires a separate filter.

Assigning users in the XML is available only when you create data filters using **manual specification**. When you assign users in the XML, you need the data filter IDs to specify the assignment, and these IDs are available only when you specify data filters using manual specification, not with automatic specification.

Details about using both methods for assigning users and groups are provided in this chapter.

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 Unica Campaign in particular (if you are setting up data filters that will be used in an application in the Unica 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 Unica 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 Unica Campaign family, these logical names must be the same as names assigned to fields in Unica Campaign. This name is used by the utility when it generates data filters.

Configuration process roadmap: creating data filters

Use this configuration process roadmap to scan the tasks required to configure data filters. The Topic column provides links to the topics that describe the tasks in detail.

Table 55. Data filter configuration process roadmap

Topic	Information
<ul style="list-style-type: none"> • Planning your data filter criteria: automatic generation (on page 191) • Planning your data filter criteria: manual generation (on page 192) 	Decide what customer data you want to secure.
Obtaining the JDBC driver for your database: automatic generation only (on page 193)	For automatic generation only: obtain the Type 4 JDBC driver that provides connectivity to the database containing the table on which you want to base your data filters.
Obtaining required information (on page 193)	Gather the required database information, and, if you plan to use the data filters with an application in the Unica Campaign family, the Unica Campaign-related information.
Creating the XML to specify data filters (on page 194)	Create the XML file that specifies the customer data used as criteria in each data filter.
Setting required data filter configuration properties (on page 194)	Set configuration properties that enable data filtering.
Populating the data filter system tables (on page 195)	Run the <code>datafilteringScriptTool</code> utility, which uses your XML to populate the Unica Platform system tables that are used for data filters.
Assigning users and groups to data filters (on page 195)	If you do not assign users and groups to data filters within the XML, use the Unica data filter user interface to perform searches for users, groups, and data filters and then select items from the search results and assign them.

Planning your data filter criteria: automatic generation

Data filter criteria are based on your customer data. Before you can define data filters, you need to decide what customer data you want to secure.

For example, you might want to restrict access to customer data based on the countries, cities, and states where your customers live. If your customer database has a table that contains country, city, and state fields, you might choose to base a group of data filters on these fields. You would then use these values when you specify your data filters.

You should be aware of the following concepts when you plan how to create data filters using automatic generation.

- **profile field** - A field whose value is considered when the data filter generation utility looks for unique combinations of values. The utility creates a data filter for each unique combination of values. When the data filter is in effect in an Unica 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 Unica application, this value is used in a WHERE clause when customer records are queried. Because the clause tests for equality, fixed fields must be defined against fields that support a finite set of distinct values.

In the example above, you would probably create a fixed field for a country, and profile fields for city and state. The data filter generation utility creates a data filter for each unique combination of values it finds in these fields.

A Unica user assigned to one or more data filters would be able to view and work with only the data belonging to the customers who live in the countries, cities, and states represented by the assigned data filter(s).

It is possible that your customer tables do not contain every value for which you want to create a data filter. For example, you might not have customers in every country and state, but might want to prepare data filters for every country and state for future use. In that case, you can reference a table that includes every country and state and use it in the GenerateDataFilters section of your XML specification. When you have finished using the utility to create your data filters, you can discard this 'dummy' table.

Planning your data filter criteria: manual generation

Data filter criteria are based on your customer data. Before you can define data filters, you need to decide what customer data you want to secure.

For example, you might want to restrict access to customer data based on the geographical sales territory to which the Unica 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 Unica user assigned to one or more data filters would be able to view and work with only the data belonging to the customers who fall within the sales territory or territories represented by the assigned data filter(s).

The data filters you create using the manual method can be assigned to users through the user interface or by making the assignments in the XML.

Obtaining the JDBC driver for your database: automatic generation only

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

1. Obtain the Type 4 JDBC driver that provides connectivity to the database containing the table on which you want to base your data filters.
2. Place the driver on the machine where Unica Platform is installed.
3. Make a note of the class name and path.

Obtaining required information

To create data filters, you need to gather information about your data and the way it is mapped in your Unica products.

For **manual specification** only: Obtain the following information.

- The physical name of the table containing the fields you want to use.
- The finite set of data in the fields you want to use for field constraints.
- If you plan to use the data filters in an application that is a member of the Unica Campaign family, obtain the names assigned in Unica Campaign to the following fields.
 - The audience fields
 - The fields you plan to use for field constraints.

For **automatic generation** only: Obtain the following information.

- For the database that contains the table you want to use in defining your data filters, the database type, the name or IP address, and the port.
- Database credentials (user name and password) that allow you to connect to the database.
- The physical name of the table containing the fields you want to use.
- The physical names of the fields you want to use for profile fields and fixed fields (fixed fields are optional).
- If you plan to use the data filters in an application that is a member of the Unica Campaign family, obtain the names assigned in Unica Campaign to the following fields.
 - The audience fields.
 - The fields you plan to use for fixed and profile fields.



Note: If you are defining data filters that will be used in an application that is a member of the Unica 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 Unica Campaign.

Creating the XML to specify data filters

Create the XML file that specifies the customer data used as criteria in each data filter. In the next step you will run a utility that populates the system tables with these specifications.

To create the data filters, the `datafilteringScriptTool` utility uses an XML representation of the data to insert entries into the Unica Platform system table database.

Here is an overview of the elements in the XML that you create.

- `<Execute Batch>` - Command that initiates the data insertion process. This is repeated several times within the XML.
- `<AddDataConfiguration>` - Defines the data configurations, which are groups of related data filters.
- `<AddLogicalFields>` - Defines the fields on which to filter, and the data type of the fields.
- `<AddDataFilter>` - When you use **manual specification**, references a defined logical field, and specifies the field constraints.
- `<GenerateDataFilters>` - When you use **automatic specification**, references the fields and the values that limit the records considered for unique combinations of values used to define a set of data filters.
- `<AddDataTable>` - Defines the relationship between logical fields and their physical tables and columns. One logical field can apply to different physical tables, which allows one filter to apply to several tables.
- `<addAudiences>` - References a defined logical field, and specifies the audience level as defined in Unica Campaign.
- `<addAudienceTableAssociations>` - Defines the relationship between an audience level and the defined table and the defined data filter configuration.
- `<AddAssignments>` - When you **create assignments within the XML rather than using the user interface**, associates individual users or groups of users with defined data filters.

For additional information, including descriptions of additional elements that are nested within the elements described above, see these topics in this chapter:

- The detailed descriptions of each element in the XML
- The XML provided in the example scenarios

Setting required data filter configuration properties

Set required configuration properties to enable data filtering.

On the **Settings & Configuration** page, navigate to the **General | Data filtering** category and set the following properties.

- Default table name
- Default audience name

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

Optional configuration property to improve data filter performance

You can turn the data filter cache on for better performance.

To improve performance, set the value of the **General | Data filtering | Enable data filter cache** property to **true**. This property specifies whether Unica Platform retrieves data filter definitions from the database or from a cache. When this value is **true**, data filter definitions are stored in the cache and the cache is updated whenever there is any change in the data filter definitions.

You must restart the Unica Platform web application after you make a change in this property value before it can take effect.

Populating the data filter system tables

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

For details on using the `datafilteringScriptTool` utility, see the full description elsewhere in this guide.



Note: If you need to delete data filters, run the `ManagerSchema_PurgeDataFiltering.sql` script, described elsewhere in this guide.

Assigning users and groups to data filters

If you do not assign users or groups within the XML that you create, use the Unica data filter user interface to perform searches for users, groups, and data filters and then select items from the search results and assign them.

Data filter XML reference

This section describes the XML elements for which you must provide values.

About the IDs in the XML

Some objects require IDs. For example, data configurations, logical fields, and data tables all require that you specify IDs. The IDs you specify must be unique within a category of object.

Some objects reference other objects using IDs. For example, tables reference logical fields. When you need to reference another object, use the ID you specified for the object.

The XML uses the following convention for ID element names. This convention helps you understand when you must create a unique ID and when you must reference another ID within the XML.

- When you must create a unique ID, the element is named `id`.
- When you must reference another object ID, the element is named for the object. For example, the ID element where you reference a logical field is named `logicalFieldId`.

Note that the IDs you assign to an object are not the IDs Unica Platform assigns to the object. The IDs you assign are used only for referencing the object within the XML.

AddDataConfiguration | dataConfiguration

This group of elements is used to define data configurations you use to group related data filters. You should create a data configuration for every set of related data filters.

Table 56. AddDataConfiguration | dataConfiguration

Element	Description	System table
<code>id</code>	Unique ID that you assign to this data configuration.	N/A
<code>name</code>	Name that you assign to this group of data filters.	Table: <code>df_config</code> Field: <code>config_name</code>

AddLogicalFields | logicalFields | LogicalField

This group of elements is used to define the logical fields corresponding to the fields in the customer table that you use to define your data filters. Create one logical field for each field from which you want to create field constraints, and one logical field for each audience.

Table 57. AddLogicalFields | logicalFields | LogicalField

Element	Description	System table
<code>id</code>	Unique ID that you assign to this logical field.	N/A
<code>name</code>	Logical name for this field or audience. If used with an application in the Unica Campaign family, must be the same as the field or audience name used in Unica Campaign.	Table: <code>df_logical_field</code> Field: <code>logical_name</code>
<code>type</code>	Data type of this field in the customer table. Allowed values are: <ul style="list-style-type: none"> • <code>java.lang.String</code> • <code>java.lang.Long</code> • <code>java.lang.Double</code> • <code>java.lang.Boolean</code> 	Table: <code>df_logical_field</code> Field: <code>type</code>

Table 57. AddLogicalFields | logicalFields | LogicalField (continued)

Element	Description	System table
	<ul style="list-style-type: none"> • java.lang.Date (The date format is month/day/year, where the month, day, and year are all expressed as numbers.) 	

GenerateDataFilters

This group of elements is used to generate data filters when you use **automatic generation**.

Table 58. GenerateDataFilters

Element	Description	System table
tableName	Physical name of the table from which you want to generate data filters, including the database schema name. If the database is case-sensitive, must match case used in the database.	Table: df_table Field: table_name
configurationName	Name of the data configuration in the <code>Ad-dDataConfiguration dataConfiguration</code> element with which this set of data filters is associated.	N/A
jdbcUrl	The URL reference for the customer database containing the table on which you want to base the data filters.	N/A
jdbcUser	The user name of an account with access to the customer database.	N/A
jdbcPassword	The password of the account with access to the customer database.	N/A
jdbcDriverClass	The name of the JDBC driver that provides connectivity to the customer database.	N/A
jdbcDriverClassPath string	The path of the JDBC driver.	N/A

GenerateDataFilters | fixedFields | FixedField

This group of elements is used to specify the optional fields and the values that limit the records considered when the data filter generation utility looks for unique combinations of values to define a set of data filters. Used only when you use **automatic generation**.

Table 59. GenerateDataFilters | fixedFields | FixedField

Element	Description	System table
<code>expression</code>	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
<code>logicalFieldName</code>	Name of the logical field in the AddLogicalFields logicalFields LogicalField element. This name appears as a label in the advanced search field in the Data Filter user interface in Unica Platform.	Table: df_logical_field Field: logical_name
<code>physicalFieldName</code>	Physical name of the field. If the database is case-sensitive, must match case used in the database.	N/A

GenerateDataFilters | profileField | ProfileField

This group of elements is used to specify fields whose unique combinations of values are used to define a set of data filters. Used only when you use **automatic generation**.

Table 60. GenerateDataFilters | profileField | ProfileField

Element	Description	System table
<code>logicalFieldName</code>	Name of the logical field in the AddLogicalFields logicalFields LogicalField element.	Table: df_logical_field Field: logical_name
<code>physicalFieldName</code>	Physical name of the field. If the database is case-sensitive, must match case used in the database.	N/A

AddDataTable | dataTable

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

Table 61. AddDataTable | dataTable

Element	Description	System table
id	Unique ID that you assign to this table.	N/A
name	Physical name of the customer table that you want to secure. If the database is case-sensitive, must match case used in the database.	Table: df_table Field: table_name

AddDataFilters | dataFilters | DataFilter

This group of elements is used to create a data filter when you use **manual specification**.

Table 62. AddDataFilters | dataFilters | DataFilter

Element	Description	System table
configId	ID of the data configuration in the AddData-Configuration dataConfiguration element with which this filter is associated.	N/A
id	Unique ID that you assign.	N/A

AddDataFilters | dataFilters | DataFilter | fieldConstraints | FieldConstraint

This group of elements is used to specify the data in a field used to define a data filter when you use **manual specification**.

Table 63. AddDataFilters | dataFilters | DataFilter | fieldConstraints | FieldConstraint

Element	Description	System table
logicalFieldId	ID of the logical field in the AddLogicalFields logicalFields LogicalField element.	N/A
expression	One item of the data in a field that is used in a WHERE clause when retrieving data for a user assigned to this filter. If the database is case-sensitive, must match case used in the database.	Table: df_fieldconstraint Field: expression

AddDataTable | dataTable | fields | TableField

This group of elements is used to map physical fields in the customer table to logical fields that you have defined.

Table 64. AddDataTable | dataTable | fields | TableField

Element	Description	System table
name	Physical name of the field in the customer table. If the database is case-sensitive, must match case used in the database.	Table: df_table_field Field: physical_name
logicalFieldId	ID of the logical field in the <code>AddLogicalFields</code> <code>logicalFields</code> <code>LogicalField</code> element.	N/A

AddAudience | audience

This group of elements is used to specify the name assigned in Unica Campaign to an audience level used in the Unica Campaign family of products.

Table 65. AddAudience | audience

Element	Description	System table
id	Unique ID that you assign to this audience.	N/A
name	Name of the audience as specified in Unica Campaign.	Table: df_audience Field: audience_name

AddAudience | audience | fields | AudienceField

This group of elements is used to specify the field or fields in your customer tables that are used as audience fields.

Table 66. AddAudience | audience | fields | AudienceField

Element	Description	System table
logicalFieldId	ID of the logical field in the <code>AddLogicalFields</code> <code>logicalFields</code> <code>LogicalField</code> element. If used with an application in the Unica Campaign family, must be the same logical name used in Unica Campaign.	N/A
fieldOrder	For future use. Set the value to 0.	N/A

addAudienceTableAssociations | addAudienceTableAssociation | audienceTableAssociation

This group of elements is used to associate pairs of audience fields and tables with data configurations. Create an association for every audience field.

Table 67. addAudienceTableAssociations | addAudienceTableAssociation | audienceTableAssociation

Element	Description	System table
audienceId	ID of the audience to be used in this association. Must be an ID value in an <code>AddAudience</code> <code>audience</code> element.	N/A
tableId	ID of the table to be used in this association. Must be an ID value in an <code>AddDataTable</code> <code>dataTable</code> element. The table must be one that contains the audience specified in the <code>audienceID</code> 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 <code>AddDataConfiguration</code> <code>dataConfiguration</code> element.	N/A

AddAssignments | assignments | AssignmentByName

You can use this group of elements to associate users or groups with data filters. Optional. You can also make these assignments through the user interface.

Table 68. AddAssignments | assignments | AssignmentByName

Element	Description	System table
namespaceId	Name of the data configuration in the <code>AddDataConfiguration</code> <code>dataConfiguration</code> element with which this set of data filters is associated.	N/A
dataObjectId	ID of the filter to be used in this association. Must be an ID value in a <code>DataFilter</code> element.	N/A
principalType	The type of assignment. <ul style="list-style-type: none"> • 1 is for assigning a data filter to an individual user • 2 is for assigning a data filter to a group of users 	Table: <code>ols_assignment</code> Field: <code>principal_type</code>

Table 68. AddAssignments | assignments | AssignmentByName (continued)

Element	Description	System table
principalName	<ul style="list-style-type: none"> • If the value used for principalType is 1, set the value to the Unica Platform login of the user you want to assign to the referenced data filter. • If the value used for principalType is 2, set the value to the name of the Unica Platform group whose members you want to assign to the referenced data filter. 	Table: ols_assignment Field: principal_id

Example: Manually specifying data filters

Jim needs to create a set of data filters based on sales territories.

In Unica 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 Unica Campaign mappings.

Table 69. Territory table fields

Fields (physical name)	Fields (name in Unica Campaign)	Data	Data type
cust_region	CustomerRegion	<ul style="list-style-type: none"> • Africa • Asia • Europe • Middle East • North America 	java.lang-String
hh_id	HouseholdID	N/A	java.lang.Long
indiv_id	IndividualID	N/A	java.lang.Long

Jim learns that the audience names used in Unica Campaign are household and individual. He notes that the Territory table contains two audience fields. The hh_id field corresponds to the household audience. The indiv_id field in the Territory table corresponds to the individual audience.

Because Jim must create one logical field for each audience, and one for the field constraint field, he knows he needs a total of three logical fields.

Jim also knows he needs to group the data filters in a data configuration. He decides to name his data configuration Territory.

Jim is now ready to create the XML.

Creating the XML

Here is the XML that Jim creates. Values based on the information he obtained are shown in `bold`.

```
<ExecuteBatch>
    <!-- ***** -->
    <!--      Data configuration      -->
    <!-- ***** -->

    <name>SeedData</name>
    <operations>
        <ExecuteBatch>
            <name>DataFilters</name>
            <operations>
                <AddDataConfiguration>
                    <dataConfiguration>
                        <id>1</id>
                        <name>Territory</name>
                    </dataConfiguration>
                </AddDataConfiguration>
            </operations>
        </ExecuteBatch>
        <!-- ***** -->
        <!--      Logical fields      -->
        <!-- ***** -->

        <AddLogicalFields>
            <logicalFields>
                <LogicalField>
                    <id>1</id>
                    <name>CustomerRegion</name>
                    <type>java.lang.String</type>
                </LogicalField>
            </logicalFields>
        </AddLogicalFields>
    </operations>
</ExecuteBatch>
```

```

<LogicalField>
  <id>2</id>
  <name>HouseholdID</name>
  <type>java.lang.Long</type>
</LogicalField>
<LogicalField>
  <id>3</id>
  <name>IndividualID</name>
  <type>java.lang.Long</type>
</LogicalField>
</logicalFields>
</AddLogicalFields>
  <!-- ***** -->
  <!-- Territory field constraints -->
  <!-- ***** -->
<AddDataFilters>
  <dataFilters>
    <DataFilter>
      <configId>1</configId>
      <id>1</id>
      <fieldConstraints>
        <FieldConstraint>
          <logicalFieldId>1</logicalFieldId>
          <expression>Africa</expression>
        </FieldConstraint>
      </fieldConstraints>
    </DataFilter>
    <DataFilter>
      <configId>1</configId>
      <id>2</id>
      <fieldConstraints>
        <FieldConstraint>
          <logicalFieldId>1</logicalFieldId>
          <expression>Asia</expression>
        </FieldConstraint>
      </fieldConstraints>
    </DataFilter>
    <DataFilter>
      <configId>1</configId>
      <id>3</id>
      <fieldConstraints>
        <FieldConstraint>

```

```

        <logicalFieldId>1</logicalFieldId>
        <expression>Europe</expression>
    </FieldConstraint>
</fieldConstraints>
</DataFilter>
<DataFilter>
    <configId>1</configId>
    <id>4</id>
    <fieldConstraints>
        <FieldConstraint>
            <logicalFieldId>1</logicalFieldId>
            <expression>Middle East</expression>
        </FieldConstraint>
    </fieldConstraints>
</DataFilter>
<DataFilter>
    <configId>1</configId>
    <id>5</id>
    <fieldConstraints>
        <FieldConstraint>
            <logicalFieldId>1</logicalFieldId>
            <expression>North America</expression>
        </FieldConstraint>
    </fieldConstraints>
</DataFilter>
</dataFilters>
</AddDataFilters>
    <!-- ***** -->
    <!-- Map physical to logical fields -->
    <!-- ***** -->
<ExecuteBatch>
    <name>addTables</name>
    <operations>
        <AddDataTable>
            <dataTable>
                <id>1</id>
                <name>Territory</name>
                <fields>
                    <TableField>
                        <name>cust_region</name>
                        <logicalFieldId>1</logicalFieldId>
                    </TableField>

```

```

    <TableField>
      <name>hh_id</name>
      <logicalFieldId>2</logicalFieldId>
    </TableField>
  <TableField>
    <name>indiv_id</name>
    <logicalFieldId>3</logicalFieldId>
  </TableField>
</fields>
</dataTable>
</AddDataTable>
</operations>
</ExecuteBatch>
  <!-- ***** -->
  <!--           Audience table associations           -->
  <!-- ***** -->
<ExecuteBatch>
  <name>addAudiences</name>
  <operations>
    <AddAudience>
      <audience>
        <id>1</id>
        <name>household</name>
        <fields>
          <AudienceField>
            <logicalFieldId>2</logicalFieldId>
            <fieldOrder>0</fieldOrder>
          </AudienceField>
        </fields>
      </audience>
    </AddAudience>
    <AddAudience>
      <audience>
        <id>2</id>
        <name>individual</name>
        <fields>
          <AudienceField>
            <logicalFieldId>3</logicalFieldId>
            <fieldOrder>0</fieldOrder>
          </AudienceField>
        </fields>
      </audience>
    </AddAudience>
  </operations>
</ExecuteBatch>

```

```

    </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 Unica 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 Unica with an account that has Admin access in Unica Platform.

He knows that groups have already been set up in Unica 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 Unica 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.

Example: Automatically generating a set of data filters

Jim needs to create a set of data filters based on countries, cities, and states.

In Unica 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 Unica 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 Unica Campaign mappings.

Table 70. Geographic table fields

Fields (Physical name)	Fields (Name in Unica Campaign)	Data	Data type
country	Country	<ul style="list-style-type: none"> • USA • France • Britain 	java.lang.String
city	City	A finite set of distinct cities	java.lang.String
state	State	A finite set of distinct states (or otherwise named regions, depending on country)	java.lang.String
hh_id	HouseholdID	N/A	java.lang.Long
indiv_id	IndividualID	N/A	java.lang.Long

Jim learns that the audience names used in Unica Campaign are household and individual. He notes that the Geographic table contains two audience fields.

- The `hh_id` field corresponds to the household audience.
- The `indiv_id` field in the Geographic table corresponds to the individual audience.

Because Jim must create one logical field for each audience, and one for each of the fixed and profile fields, he knows he needs a total of five logical fields.

Jim also knows he needs to group the data filters in a data configuration. He decides to name his data configuration Geographic.

Jim is now ready to create the XML.

Creating the XML

Here is the XML that Jim creates. Values based on the information he obtained or decided to use are shown in `bold`.

```
<ExecuteBatch>
    <!-- ***** -->
    <!--      Data configuration      -->
    <!-- ***** -->
```

```

<name>SeedData</name>
<operations>
  <ExecuteBatch>
    <name>DataFilters</name>
    <operations>
      <AddDataConfiguration>
        <dataConfiguration>
          <id>1</id>
          <name>Geographic</name>
        </dataConfiguration>
      </AddDataConfiguration>
    </operations>
  </ExecuteBatch>
  <!-- ***** -->
  <!--      Logical fields      -->
  <!-- ***** -->
  <AddLogicalFields>
    <logicalFields>
      <LogicalField>
        <id>1</id>
        <name>Country</name>
        <type>java.lang.String</type>
      </LogicalField>
      <LogicalField>
        <id>2</id>
        <name>City</name>
        <type>java.lang.String</type>
      </LogicalField>
      <LogicalField>
        <id>3</id>
        <name>State</name>
        <type>java.lang.String</type>
      </LogicalField>
      <LogicalField>
        <id>4</id>
        <name>HouseholdID</name>
        <type>java.lang.Long</type>
      </LogicalField>
      <LogicalField>
        <id>5</id>
        <name>IndividualID</name>
        <type>java.lang.Long</type>
    </logicalFields>
  </AddLogicalFields>

```

```

</LogicalField>
</logicalFields>
</AddLogicalFields>
  <!-- ***** -->
  <!--      Generate data filters      -->
  <!-- ***** -->
<GenerateDataFilters>
  <!-- ***** -->
  <!-- Specify the table to be scanned for unique combinations -->
  <!-- of values from which data filters will be defined. -->
  <!-- ***** -->
  <tableName>Geographic</tableName>
  <!-- ***** -->
  <!-- Identify the data configuration with which -->
  <!-- generated data filters will be associated. -->
  <!-- ***** -->
  <configurationName>Geographic</configurationName>
  <!-- Specify the data source connection information. -->
  <jdbcUrl>
    jdbc:sqlserver://localhost:1433;databaseName=Customers
  </jdbcUrl>
  <jdbcUser>sa</jdbcUser>
  <jdbcPassword>myPassword</jdbcPassword>
  <jdbcDriverClass>
    com.microsoft.sqlserver.jdbc.SQLServerDriver</jdbcDriverClass>
  <jdbcDriverClassPath>
    <string>C:\tools\Java\MsJdbc\sqljdbc.jar</string>
  </jdbcDriverClassPath>
  <!-- ***** -->
  <!--      Specify the fixed fields      -->
  <!-- ***** -->
  <fixedFields>
    <FixedField>
      <expression>USA</expression>
      <logicalFieldName>Country</logicalFieldName>
      <physicalFieldName>country</physicalFieldName>
    </FixedField>
    <FixedField>
      <expression>France</expression>
      <logicalFieldName>Country</logicalFieldName>
      <physicalFieldName>country</physicalFieldName>
    </FixedField>
  </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>
    <!-- ***** -->
    <!-- Audience table associations -->
    <!-- ***** -->
</ExecuteBatch>
<name>addAudiences</name>
<operations>
<AddAudience>
    <audience>
        <id>1</id>
        <name>household</name>
        <fields>
            <AudienceField>
                <logicalFieldId>4</logicalFieldId>
                <fieldOrder>0</fieldOrder>
            </AudienceField>
        </fields>
    </audience>
</AddAudience>
<AddAudience>
    <audience>
        <id>2</id>
        <name>individual</name>
        <fields>
            <AudienceField>
                <logicalFieldId>5</logicalFieldId>
                <fieldOrder>0</fieldOrder>
            </AudienceField>
        </fields>
    </audience>
</AddAudience>

```

```

    </operations>
  </ExecuteBatch>
  <!-- ***** -->
  <!-- Associate table-audience pairs -->
  <!-- with data configuration -->
  <!-- ***** -->
  <ExecuteBatch>
    <name>addAudienceTableAssociations</name>
    <operations>
      <AddAudienceTableAssociation>
        <audienceTableAssociation>
          <audienceId>1</audienceId>
          <tableId>1</tableId>
          <configId>1</configId>
        </audienceTableAssociation>
      </AddAudienceTableAssociation>
      <AddAudienceTableAssociation>
        <audienceTableAssociation>
          <audienceId>2</audienceId>
          <tableId>1</tableId>
          <configId>1</configId>
        </audienceTableAssociation>
      </AddAudienceTableAssociation>
    </operations>
  </ExecuteBatch>
</operations>
</ExecuteBatch>

```

Populating the system tables

Jim has named his data filter XML file `geographicDataFilters.xml` and saved it in `tools/bin` directory under his Unica 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 Unica Platform with an account that has Admin access in Unica Platform.

He knows that groups have already been set up in Unica 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 Unica 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.

About assigning users and groups in the XML

You can assign users or groups to data filters in the XML, as an alternative to doing this through the user interface. Assigning users and groups to data filters in the XML is available only when you use manual specification to create the data filters.

You can use a wild card, `#user_login#`, that automatically creates data filters based on the user's Unica Platform login name.

You use the `AddAssignments` XML element block to associate users or groups with your data filters.

Scenario used in the example

The example uses the following scenario.

An organization uses Unica Collaborate and wants to create data filters that allow field marketers to see only the customers in the region to which they are assigned. Thus, each user requires his or her own data filter.

In Unica Collaborate the list display and the form templates are set up based on region. This configuration is described in more detail in the Unica Collaborate Administrator's Guide.

The audience level is Customer.

The data filters are created against four tables in the `exampleSchema` database, as described in the following table.

Table 71. Tables and fields used in the examples

Table	Fields
<code>exampleSchema.Corporate_Lists</code>	<p>UserID, State, and RegionID</p> <p>This is the list display table set up in Unica Collaborate. The UserID column contains the Unica Platform login names of the field marketers. This table associates field marketer Unica Platform login names with their assigned region.</p>

Table 71. Tables and fields used in the examples (continued)

Table	Fields
<code>exampleSchema.customer_contact</code>	Indiv_ID, Region_ID, and State fields for customers
<code>exampleSchema.lkup_state</code>	A lookup table for the <code>state_name</code> field
<code>exampleSchema.lkup_region</code>	A lookup table for the <code>region_id</code> field

Example: Using the wild card to assign group members to data filters

To create a separate data filter for each member of a specified group, you do the following.

- Create logical fields as usual.
- Create a single data filter with the wild card `#user_login#` in the `expression` element.
- Under the `AssignmentByName` element, set the `principalType` to 2, set the `principalName` element to the group name, and set the `dataObjectId` element to the ID of the wild card data filter.
- Create audience associations as usual.

The following XML illustrates this method, using the scenario described above.

```
<ExecuteBatch>
  <!-- ***** -->
  <!--      Data configuration      -->
  <!-- ***** -->

  <name>SeedData</name>
  <operations>
    <ExecuteBatch>
      <name>DataFiltering</name>
      <operations>
        <AddDataConfiguration>
          <dataConfiguration>
            <id>1</id>
            <name>collaborate</name>
          </dataConfiguration>
        </AddDataConfiguration>
      </operations>
    </ExecuteBatch>
    <!-- ***** -->
    <!--      Logical fields      -->
    <!-- ***** -->

    <AddLogicalFields>
      <logicalFields>
```



```

<LogicalField>
  <id>1</id>
  <name>Customer_ID</name>
  <type>java.lang.String</type>
</LogicalField>

<LogicalField>
  <id>2</id>
  <name>AudienceLevel</name>
  <type>java.lang.String</type>
</LogicalField>

<LogicalField>
  <id>3</id>
  <name>UserID</name>
  <type>java.lang.String</type>
</LogicalField>

<LogicalField>
  <id>4</id>
  <name>State_code</name>
  <type>java.lang.String</type>
</LogicalField>

<LogicalField>
  <id>5</id>
  <name>Region</name>
  <type>java.lang.Long</type>
</LogicalField>
</logicalFields>
</AddLogicalFields>
<!-- ***** -->
<!--      Wild card data filter      -->
<!-- ***** -->
<AddDataFilters>
  <dataFilters>
    <DataFilter><
      <configId>1</configId>
      <id>1</id>
      <fieldConstraints>
        <FieldConstraint>
          <logicalFieldId>3</logicalFieldId>

```

```

        <expression>#user_login#</expression>
    </FieldConstraint>
</fieldConstraints>
</DataFilter>
</dataFilters>
</AddDataFilters>
<!-- ***** -->
<!--          Add data tables          -->
<!-- ***** -->

<ExecuteBatch>
  <name>addTables</name>
  <operations>
    <!-- ***** -->
    <!--          Table exampleSchema.Corporate_Lists          -->
    <!-- ***** -->
    <AddDataTable>
      <dataTable>
        <id>1</id>
        <name>exampleSchema.Corporate_Lists</name>
        <fields>
          <TableField>
            <tableId>1</tableId>
            <name>UserID</name>
            <logicalFieldId>3</logicalFieldId>
          </TableField>
          <TableField>
            <tableId>1</tableId>
            <name>State</name>
            <logicalFieldId>4</logicalFieldId>
          </TableField>
          <TableField>
            <tableId>1</tableId>
            <name>Region_ID</name>
            <logicalFieldId>5</logicalFieldId>
          </TableField>
        </fields>
      </dataTable>
    </AddDataTable>
    <!-- ***** -->
    <!--          Table exampleSchema.customer_contact          -->
    <!-- ***** -->

```

```

<AddDataTable>
  <dataTable>
    <id>2</id>
    <name>exampleSchema.customer_contact</name>
    <fields>
      <TableField>
        <tableId>2</tableId>
        <name>Indiv_ID</name>
        <logicalFieldId>1</logicalFieldId>
      </TableField>
      <TableField>
        <tableId>2</tableId>
        <name>Region_ID</name>
        <logicalFieldId>5</logicalFieldId>
      </TableField>
      <TableField>
        <tableId>2</tableId>
        <name>State</name>
        <logicalFieldId>4</logicalFieldId>
      </TableField>
    </fields>
  </dataTable>
</AddDataTable>
<!-- ***** -->
<!--           Table exampleSchema.lkup_state           -->
<!-- ***** -->
<AddDataTable>
  <dataTable>
    <id>3</id>
    <name>example.schema.lkup_state</name>
    <fields>
      <TableField>
        <tableId>3</tableId>
        <name>state_name</name>
        <logicalFieldId>4</logicalFieldId>
      </TableField>
    </fields>
  </dataTable>
</AddDataTable>
<!-- ***** -->
<!--           Table exampleSchema.lkup_region           -->
<!-- ***** -->

```

```

    <AddDataTable>
      <dataTable>
        <id>4</id>
        <name>exampleSchema.lkup_region</name>
        <fields>
          <TableField>
            <tableId>4</tableId>
            <name>Region_ID</name>
            <logicalFieldId>5</logicalFieldId>
          </TableField>
        </fields>
      </dataTable>
    </AddDataTable>
  </operations>
</ExecuteBatch>

<!-- ***** -->
<!-- Audience table associations -->
<!-- ***** -->

<ExecuteBatch>
  <name>addAudiences</name>
  <operations>
    <AddAudience>
      <audience>
        <id>1</id>
        <name>Customer</name>
        <fields>
          <AudienceField>
            <logicalFieldId>2</logicalFieldId>
            <fieldOrder>0</fieldOrder>
          </AudienceField>
        </fields>
      </audience>
    </AddAudience>

    <AddAudience>
      <audience>
        <id>2</id>
        <name>default</name>
        <fields>
          <AudienceField>
            <logicalFieldId>2</logicalFieldId>
            <fieldOrder>0</fieldOrder>
          </AudienceField>
        </fields>
      </audience>
    </AddAudience>
  </operations>
</ExecuteBatch>

```

```

        </AudienceField>
      </fields>
    </audience>
  </AddAudience>
</operations>
</ExecuteBatch>

<ExecuteBatch>
  <name>addAudienceTableAssociations</name>
  <operations>
    <AddAudienceTableAssociation>
      <audienceTableAssociation>
        <audienceId>1</audienceId>
        <tableId>1</tableId>
        <configId>1</configId>
      </audienceTableAssociation>
    </AddAudienceTableAssociation>

    <AddAudienceTableAssociation>
      <audienceTableAssociation>
        <audienceId>1</audienceId>
        <tableId>2</tableId>
        <configId>1</configId>
      </audienceTableAssociation>
    </AddAudienceTableAssociation>

    <AddAudienceTableAssociation>
      <audienceTableAssociation>
        <audienceId>2</audienceId>
        <tableId>3</tableId>
        <configId>1</configId>
      </audienceTableAssociation>
    </AddAudienceTableAssociation>

    <AddAudienceTableAssociation>
      <audienceTableAssociation>
        <audienceId>2</audienceId>
        <tableId>4</tableId>
        <configId>1</configId>
      </audienceTableAssociation>
    </AddAudienceTableAssociation>
  </operations>
</ExecuteBatch>

```

```

    </operations>
  </ExecuteBatch>

  <!-- ***** -->
  <!--           Link filters (dataObjectId) to group           -->
  <!-- ***** -->

  <AddAssignments>
    <assignments>
      <AssignmentByName>
        <namespaceId>1</namespaceId>
        <dataObjectId>1</dataObjectId>
        <principalType>2</principalType>
        <principalName>FieldMarketer</principalName>
      </AssignmentByName>
    </assignments>
  </AddAssignments>
</operations>
</ExecuteBatch>

```

About assigning user and groups through the user interface

You can assign users and groups to data filters on the **Settings > Data filters** pages.

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

- The data filters must be set up in Unica Platform system tables.
- You must log in as a user with the Unica Platform permission **Administer data filters page**. The pre-configured Unica Platform **AdminRole** role has this permission.

Advanced search

Unica Platform provides a user interface for assigning users and groups to data filters. This user interface relies on an advanced search feature to obtain lists of users, groups, and data filters. You can select users and groups from these lists and assign them to data filters that you select.

Data filter search

The search feature for data filters provides search criteria that are the same as the criteria specified when the data filters were set up. For example, suppose a set of data filters is based on a field containing the following data relating to sales territories.

- Africa
- Asia
- Europe

- Middle East
- North America

The data filter advanced search would provide this data in a drop-down list from which you can select when searching for data filters.

User and group search

The advanced search feature for users and groups provides a text field where you can enter text for the search to match.

When a tab containing the user and group advanced search first loads, there is a wildcard (*) in both the User and Group text fields. A search performed using this wildcard returns all records.

If you delete the wildcard and do not enter any other text, leaving the field blank, no records are returned. For example, if you perform a search with the User text field blank and an asterisk in the Group text field, only groups would be listed in the results.

On the View Assignments tab, if you leave both the User and Group text fields blank, no records are returned regardless of what data filter criteria are selected.

When you enter text in the field, the search matches the characters you enter in the text field, in the order you enter them. For example, to obtain a group named North America, you could enter any letter or group of letters (in order) that occurs in the name. You would obtain North America in the results if you entered "north" or "h", but not if you entered "htron."

The search is not case-sensitive. That is, "North" is the same as "north."

Viewing assigned data filters

Use this procedure to view assigned data filters

1. Log in to Unica as a user with the Unica Platform AdminRole role and click **Data Filtering**.

The Data filters page displays.

2. Click **View assigned data filters**.
3. Perform an advanced search for assigned data filters to obtain search results.

A list of data filters that meet the criteria is displayed.

Assigning users and groups to data filters

Use this procedure to assign users and groups to data filters.

1. Log in to Unica as a user with the Unica Platform AdminRole role and click **Settings > Data filters**.

The Data filters page displays.

2. Click **Assign users or groups**.
3. Perform an advanced search for data filters to obtain a list of data filters.
4. Perform an advanced search for the users, groups, or both to obtain a list of users and groups.
5. From your search results lists, select data filters and the users and groups you want to assign to them.
6. Click **Assign**.

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

Removing data filter assignments

Use this procedure to remove data filter assignments.

1. Log in to Unica as a user with the Unica Platform AdminRole role and click **Settings > Data filters**.

The Data Filters page displays.

2. Click **View assigned data filters**.
3. Perform an advanced search for assigned data filters to obtain search results from which you want to select.
4. From your search results list, select the data filters whose assignments you want to delete.
5. Click **Unassign**.

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

Adding data filters after the initial set has been created

You can continue to add data filters after you have created the initial set. For example, you might create a set of data filters based on countries and their city/state combinations, and later decide to create another set based on zip codes.

You can obtain the XML for additional data filters in either of the following ways.

- Modify your original XML file to add new filters. When you seed the database using the `dataFilteringScriptTool` utility, the Unica Platform creates only the new data filters.
- Create an XML file specifying new data filters. When you seed the database using the `dataFilteringScriptTool` utility, existing data filters are not deleted.

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

Audit event tracking in Unica

You can configure which audit events are tracked and assign a severity level to each tracked event.

Two kinds of audit events are tracked:

- Security related events such as changes to user status, group memberships, and permissions
- Changes to Unica configuration properties that are managed on the **Settings > Configuration** page

The audit event information is independent of the system log, and configuration you perform for the system log does not affect audit event tracking.

The Audit Events report provides a convenient way to view the tracked events. You can configure the content of the report, filter the information shown in the report, and export report data.

You must have the AdminRole or PlatformAdminRole role in Unica Platform to configure the Audit Events report and audit backups or to view the report.

Limitations on audit event tracking

If you configure tracking for configuration property audit events, these changes are tracked only when they are performed using the **Settings > Configuration** page.

For example, the following configuration property changes are not tracked.

- Changes made using Unica Platform utility `configTool`
- Changes made during installation and upgrade of Unica products

Also, when you manually add default users, roles, and permissions for Unica Platform and Unica Campaign using the Unica Platform `populateDB` utility, these changes are not tracked.

Legacy audit events

Previous releases of Unica Platform saved audit events in the Unica Platform system tables, although no report was available. If you upgrade from a version earlier than 9.1.1, the Audit Events report includes these legacy events.

Legacy audit events are displayed in the report as follows.

- The **Severity** column contains **No severity (Legacy)** to indicate that these audit records were stored before the audit report was available.
- In an environment with just one partition, the **Partition** column contains the ID of the default partition.
- In a multi-partition environment, the **Partition** column contains **-1 (Legacy)** to indicate that the partition to which the event belongs cannot be determined.

The following legacy events may appear after your upgrade.

- User authentication succeeded.
- User authentication failed.
- Authentication failed because a user attempted to log in with too many concurrent sessions.
- User logged off and the corresponding session ended.
- User's password changed.

Legacy audit events are visible in reports only when you access the report with an account that has the PlatformAdminRole role in Unica Platform. The pre-defined `platform_admin` user has this role.

Retroactive changes

If the first name, last name, or email address of a user account is changed, all audit events tracked for this user reflect the changes. This is true even for events tracked before the user profile changes were made.

Permissions for viewing the Audit Events report in a multi-partition environment

In a multi-partition environment, the partition membership of the administrator viewing the Audit Events report affects the events that are included when the administrator views the report.

For all audit events except configuration events, the report shows only those events that occurred in the partition of the administrator viewing the report. Events that occurred in other partitions are not shown in the report.

The exception is administrators with the PlatformAdminRole role, who can see events that occur in all partitions.

All configuration events are visible to all administrators who can view the report.

Enabling and disabling event auditing

By default, event auditing is disabled. To enable event auditing, you set the **Unica Platform | Audit Events | Is Event Auditing enabled** configuration property to True.

This configuration property affects only the audit events listed under **Unica Platform | Audit Events** on the Configuration page. The events tracked in the system log are not affected.

You can disable event auditing at any time by setting the value of the **Is Event Auditing enabled** configuration property to False.

The Audit Events report does not include the events controlled by the **Is Event Auditing enabled** property that occurred during any period when the property was set to **False**.

Configuring which audit events appear in the report

To specify the audit events that are available in the audit report and their severity, you set properties on the **Settings > Configuration** page.

1. Go to the **Settings > Configuration** page and expand the **Unica Platform | Audit Events | Audit Events Configuration** category.
2. Select the events that you want to track.

The tracked events are available for inclusion in the audit report.

3. Expand the **Unica Platform | Audit Events | Audit events severity configuration** category and click **Edit settings**.
4. Specify the severity level that you want to assign to each of the tracked events.

Select from the following options.

- No severity
- Informational
- Warning
- Critical

The specified severity appears in the audit report, and can be used in filtering the report.

If you want to track the user session event **Record login and logout events for members of the HighSeverityAccounts group**, add the users whose login and logout events you want to track to the **highSeverityAccounts** group. You do this on the **Settings > User groups** page.

This group is created automatically in the default partition during installation. In a multi-partition environment, this group is created automatically when you create a new partition using the Unica Platform `partitionTool` utility.

Modifying the audit report content and display

You can add and remove events and columns, rearrange and sort the columns, set the time span, specify which tracked events are shown in the report, and filter the information.

When you open the audit report without setting any report parameters, the following default settings are used.

- All of the events selected on the **Settings > Configuration** page under the **Unica Platform | Audit Events | Audit Events Configuration** category are shown, on multiple pages if necessary.
- No date criteria are applied.
- Events are sorted as follows: Event Date/Time (Descending), Login Name(Ascending), Severity Level (Ascending)

Use this procedure to modify these settings.

1. Go to **Analytics > Platform**.
2. To change the content of the report, do the following.

- a. Click the **Report Parameters** button.

The Report Parameters window opens.

- b. Complete the fields.

To set the sort order in the report, you can select from pre-defined sort orders in this window. You can also click the column headers in the report to sort on those columns.

- c. Click **Next** to move to a page where you can select which events are shown in the report.

- d. Click **Save and Close** to apply your selections to the report.

3. To filter the report, enter text or numbers in the **Filter** field and click the **Filter** button.

The report displays only those events that contain the filter characters in any of the columns displayed in the report.

To clear the filter, click the **X** in the Filter field.

Fields in the Report Parameters window

Use the fields on the Report Parameters page to configure the way the audit report is displayed.

Table 72. Fields in the Report Parameters window

Field	Description
Sort	Select a sort order from the drop-down menu. Various combinations of column sorting are listed, along with whether the sort is in descending or ascending order. You can also sort columns using controls on the report page.
Time Period	Select from pre-defined time periods in the drop-down list, or enter start and end dates for a custom range.
Events	Select the optional events that you want to include in the report. For an event to be available in the report, it must be selected in the Unica Platform Audit Events Audit Events Configuration category on the Settings > Configuration page.
Columns	Use the Add and Remove buttons to specify the optional columns you want to appear in the report.

Fields and buttons in the Audit Events report

Fields in the Audit Events report provide details about system and user events in Unica.

Table 73. Fields and buttons in the Audit Event report




Field or button	Description
Filter	To filter the report, enter text or numbers in the Filter field and click the button. The report displays only those events that contain the filter characters in any of the columns displayed in the report.
 Report parameters	Click to open a window where you can change the columns displayed in the report, set a time period, and select from pre-defined sort orders.
 Export	Click to open a window where you can export the report in CSV or text format.

Table 73. Fields and buttons in the Audit Event report (continued)

Field or button	Description
 Refresh	Click to refresh the report data.
Default fields	
Event date/time (short)	The date and time of the event on the server where Unica Platform is deployed.
Event name	The tracked event. Events that are tracked are specified on the Settings > Configuration page.
Event details	Details about the tracked event. When a link is present, you can click it to see full details.
Login name	The login name of the user who performed the action.
Last name, first name	The first and last name of the user who performed the action.
Severity	The severity assigned to the event on the Unica Platform Audit Events Audit Events Severity Configuration page.
Optional fields set in the Report Parameters window	
Browser	The browser used by the person who performed the action.
Host name	The name or IP address of the machine from which the action was performed.
Request from	The URI where the HTTP request originated.
Event date/time (long)	The date and time of the event on the server where Unica Platform is deployed, including the time zone.
User email	The email address of the user who performed the action.
Partition	The partition membership of the user who performed the action.

Archived audit events

You can configure backups of audit events by setting the value of configuration properties in the **Unica Platform | Audit Events | Audit Events Configuration** category on the **Settings > Configuration** page.

The archived data is stored in the `USM_AUDIT_BACKUP` table and can be included in the Audit Events report when you set a custom date range that includes data from the archive. Loading a report that includes archived data can take longer than loading a report that does not include archived data.

The system posts a notification when an audit backup process completes. You can also configure a group of users who receive these notifications in emails.

Set the following properties to configure audit event backups.

- **Enable Audit Backup**
- **Archive data after the number of days specified here**
- **Keep Audit records in primary for number days specified here**
- **Archive start time**
- **Name of group to receive audit backup notifications**

Configuring audit backup notifications

To notify users of the status of audit event backup, make them members of a group that you specify in a configuration property.

1. Determine the group whose members you want to receive email notifications of audit data backups.

You can use an existing group or create a new one on the **Settings > User groups** page.

You can specify only a single group to receive notifications.

2. Go to the **Settings > Configuration** page and expand the **Unica Platform | Audit Events | Audit events configuration** category.
3. Set the value of the **Name of group to receive audit backup notifications** property to the name of the group you selected.
4. Add the users who should receive notifications to the group.
5. The users who you have added to the group must subscribe to the notifications on the **Settings > Users** page.

An administrator can do this for each user, or you can inform the users that they must go to their account, click **Notification subscriptions**, and subscribe to **Audit backup** notifications.

Each time the system backs up audit data, an email notification and user interface notification is generated for the members of the group you specified, if they have subscribed to Audit Event notifications.

Exporting the Audit Events report

You can export the security audit report to a text or comma separated file.

1. Go to **Analytics > Marketing Platform**.
2. Click the **Export** button.
3. In the Audit Report Export window, enter a name for your exported report, and select the export format.

The format options are:

- **CSV** (a comma separated list that Microsoft™ Excel can open)
- **TXT** (text)

If you select text format, you also choose the separator. Options are:

- #
- |
- **TAB**

4. Click **Export**, specify whether you want to open or save the exported report, and then close the export window.

Optimizing the export of the Audit Events report for large event volumes

If you want to export large audit event reports, for example, reports containing more than 65,000 records matching your audit event report filter criteria, the export can time out. To circumvent this problem, perform the following procedure.

This procedure applies when you use Internet Explorer to access the Audit Event report.

1. Edit the Windows™ registry as follows.

a. Open the Windows™ registry editor and navigate to `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings`.

b. If a DWORD entry named `ReceiveTimeout` does not exist, create one.

To create a new DWORD entry, do the following.

- Right-click on `Internet Settings` and select **New > DWORD (32-bit) Value**.
- Enter `ReceiveTimeout` as the name for the new entry.

c. Give the existing or new `ReceiveTimeout` entry a value as follows.

- Right-click on the `ReceiveTimeout` entry and select **Modify**.
- Under **Base**, select **Decimal**.
- Specify the timeout interval in milliseconds.

For example, to specify 3 hours, you would enter 10800000, which is 180 minutes * 60 seconds * 1000.

2. Configure Internet Explorer as follows.

- a. Select **Tools > Internet Options** and click the Security tab.
- b. Select the zone where you access Unica Platform. For example, Trusted Sites.
- c. Click **Custom level**.
- d. Under **Downloads**, enable **Automatic prompting for file downloads**.
- e. Restart Internet Explorer.

The Unica Platform system log

You should check the system log first if the Unica Platform application malfunctions. The system log is independent of the security audit information, which is stored in the system tables. While the system log tracks some of the same information contained in the security audit reports, it also contains information useful in troubleshooting Unica Platform.

The system log contains the following information.

- Configuration information and all errors and debugging information for the Unica Platform.
- A record of key events as they occur on the Unica Platform server (requests, grants, revokes, and failures).

About the configuration settings displayed in the system log



Note: If a problem occurs when the system attempts to write to the system log file, the system writes to stdout (command line) instead of to a file.

System log entry format

The system log entries are in the following format.

Timestamp | Event severity level | Message

- **Timestamp** – The time the event occurred.
- **Event Severity Level** – The logging level of the event.
- **Message** – Description of the event. If the entry is a request to the server, the message typically contains the function called by the request. Response entries record the results of the requests.

System log configuration

You configure the system log using the `log4j.properties` file, located by default in the `conf` directory under your Unica Platform installation. Changes to this file go into effect within 60 seconds after the file is saved.



Note: You configure the system log using the `log4j.xml` file, located by default in the `conf` directory under your Unica Platform installation. Changes to this file go into effect within 60 seconds after the file is saved.

Configuration you perform on the system log does not affect security audit reports.

Default system log settings

By default, the system log is configured as follows:

- Log file name: `platform.log`
- Log directory: `Unica/Platform/logs`
- Log level: `WARN`
- Number of backups: 10
- 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 Unica Platform data store.

Logging level settings for the whole Unica Platform system

You can change the logging level for all components of Unica Platform by .uncommenting the desired line in the Examples section of the file. To uncomment a line, remove the `<userinput>#</userinput>` character at the beginning of the line. If you make this change, be sure to add the `<userinput>#</userinput>` symbol to the beginning of the line specifying the previous logging level.



Note: You can change the logging level for all components of Unica Platform by modifying log level in Root tag defined under Loggers tag.

Setting logging levels for Unica Platform components

You can set the logging level in the system log for specific components of the Unica 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.



Note: To debug a specific module, remove the `<!--` symbol at the beginning of each `<Logger>` tag and `-->` at the end of each `<Logger>` tag of the module in the log4j.xml 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>.
- See comments in the `log4j.xml` file.
- See <https://logging.apache.org/log4j/2.x/manual/configuration.html>



Note: Users can disable warnings from JDBC by setting the following property

`hibernate.jdbc.log.warnings=false` in `platform_home/tools/bin/jdbc.properties` file.

Enabling single-user logging

You can enable single-user logging by configuring logging to use the XML file and then editing the XML file.

Logging is configured using one of two files: `log4j.properties` or `log4j.xml`. By default, the `log4j.properties` file is used.

You can enable per-user logging by configuring logging to use the XML file and then editing the XML file. If Unica Platform is configured in a cluster deployment, copy the XML file to each node.

- You can enable single-user logging by editing the XML file.
- Logging is configured using `log4j.xml`, which is the default configuration file.
- If Unica Platform is configured in a cluster deployment, copy the XML file to each node.

With XML logging enabled, a thread is created that periodically checks if the XML configuration file has been created or modified. If a change or file creation is detected, then the XML file is read to configure log4j. The polling interval is 60 seconds.

1. Configure logging to use log4j.xml by setting the following JVM parameter.

```
-DENABLE_PLATFORM_LOG4J_XML_LOGGING=true
```

The value must be set to true to enable per-user logging.

If Unica Platform is configured in a cluster deployment, set this JVM parameter in each node of the cluster.

2. To specify the user account to be logged in per-user logging, edit the log4j.xml file and add the users in the filter tag. The logs for the users that are added in the filter tag.
 - You can add multiple tags in the `log4j.xml` file to create separate log files for specific users. You must add a new appender for each new user specific log file.
 - By default, the log file is created in the `Platform_Home /Platform/logs` folder and is named as `platform.log`. You can specify a different valid path and file name. You must specify the absolute or complete path to generate the log files in the respective folders.
 - If both user specific logs and logs for all users are required, add an appender tag with a new name and without the filter tag defined. The appender must have a unique name.
 - Add a corresponding entry under the root tag for this new appender.
3. If Unica Platform is configured in a cluster deployment, copy the edited XML file to each node of the cluster.

You can use a command like the one shown in the following example.

```
-DPLATFORM_LOG4J_XML_FILE=log4j_node1.xml
```

The `log4j_node1.xml` file is a copy of the `log4j.xml` file. You can use any name for the copied file. Log file is also created with this new name like `log4j_node1.log` automatically instead of default name `platform.log`.

Consider the following example where the logs are collected for the user `asm_admin` and for all other users.

```
<appender name="Console" class="org.apache.log4j.ConsoleAppender">
<param name="ImmediateFlush" value="true"/>
<layout class="org.apache.log4j.PatternLayout">
<param name="ConversionPattern" value="%-5p %c - %m%n"/>
</layout>
<filter class="com.unica.manager.logger.UserMatchFilter">
<param name="StringToMatch" value="asm_admin" />
</filter>
</appender>
<appender name="Console" class="org.apache.log4j.ConsoleAppender">
<param name="ImmediateFlush" value="true"/>
<layout class="org.apache.log4j.PatternLayout">
```

```

<param name="ConversionPattern" value="%-5p %c - %m%n"/>
</layout>

<filter class="com.unica.manager.logger.UserMatchFilter">
<param name="StringToMatch" value="asm_admin" />
</filter>
</appender>
</appender>

<!-- <logger name="com.unica.manager.configuration.ConfigurationManager">
<level value="TRACE"/>
</logger>

<logger name="com.unica.suite.scheduler.server.manager.TaskManager">
<level value="DEBUG"/>
</logger>

<logger name="org.hibernate.util.JDBCExceptionReporter">
<level value="ERROR"/>
</logger>
-->

<root>
<level value="WARN"/>
<appender-ref ref="System"/>
<appender-ref ref="Console"/>
<appender-ref ref="SystemAllUsers"/>
</root>

```

1. To specify the user account to be logged in per-user logging, edit the `log4j.xml` file and uncomment `RollingFile` tag with name `UserLogAppender`. Add the `userid` in the filter tag. The logs for the user that is added in the filter tag are saved in the file that is mentioned in that appender. Set below JVM parameter if not already set,

```
-DUNICA_PLATFORM_HOME= <platform_home_directory_path>
```

- You can add multiple tags in the `log4j.xml` file to create separate log files for specific users. You must add a new appender for each new user specific log file.
 - By default, the log file is created in the `Platform_Home /Platform/logs` folder and is named as `platform.log`. You can specify a different valid path and file name. You must specify the absolute or complete path to generate the log files in the respective folders.
 - If both user specific logs and logs for all users are required, add an appender tag with a new name and without the filter tag defined. The appender must have a unique name.
 - Add a corresponding entry under the root tag for this new appender.
2. If Unica Platform is configured in a cluster deployment, copy the edited XML file to each node of the cluster.

You can use a command like the one shown in the following example.

```
-DPLATFORM_LOG4J_XML_FILE=log4j_node1.xml
```

The `log4j_node1.xml` file is a copy of the `log4j.xml` file. You can use any name for the copied file. Log file is also created with this new name like `log4j_node1.log` automatically instead of default name `platform.log`.

Consider the following example where the logs are collected for the user `asm_admin` and for all other users.

```
<?xml version="1.0" encoding="UTF-8"?>
<Configuration packages="com.unica.manager.logger" monitorInterval="60">
  <Appenders>

  <!-- Console Log Appender -->

  <Console name="CONSOLE_LOG" target="SYSTEM_OUT" immediateFlush="true">
    <PatternLayout pattern="%-5p %c - %m%n"/>
  </Console>

  <!-- System Log Appender -->
  <!-- The following section is for logs for all the user-->
  <RollingFile name="SYS_LOG" fileName="${sys:UNICA_PLATFORM_LOG_FILE}"
filePattern="${sys:UNICA_PLATFORM_LOG_FILE}.%d{yyyy-MM-dd}-%i"
immediateFlush="true" append="true" >
  <PatternLayout pattern="%d{DATE} - %-5p - %m%n" />
  <Policies>
    <TimeBasedTriggeringPolicy interval="1" modulate="true"/>
    <SizeBasedTriggeringPolicy size="10 MB" />
  </Policies>
  <DefaultRolloverStrategy max="10"/>
</RollingFile>

  <!-- The following section is for user specific logs for the user asm_admin-->

  <RollingFile name="UserLogAppender" fileName="${sys:UNICA_PLATFORM_HOME}/logs/asm_admin.log"
filePattern="${sys:UNICA_PLATFORM_HOME}/logs/asm_admin.log.%d{yyyy-MM-dd}"
immediateFlush="true" append="true" >
  <PatternLayout pattern="%d{yyyy-MM-dd HH:mm:ss} [%X{user}] %-5p %F.%M:%L: %m%n" />
  <Policies>
    <SizeBasedTriggeringPolicy size="10 MB" />
  </Policies>
  <DefaultRolloverStrategy max="10"/>
  <UserMatchFilter user="asm_admin" onMatch="ACCEPT" onMismatch="NEUTRAL"/>
</RollingFile>
```

```

</Appenders>
<Loggers>

<Root level="WARN" includeLocation="true">
  <AppenderRef ref="SYS_LOG"/>
  <AppenderRef ref="CONSOLE_LOG"/>
  <!-- <AppenderRef ref="UserLogAppender"/> -->
</Root>

  <!-- <Logger name="com.unicacorp" level="INFO"> -->
  <!-- <AppenderRef ref="UserLogAppender"/> -->
  <!-- </Logger> -->

  <!-- <Logger name="com.unica" level="INFO"> -->
  <!-- <AppenderRef ref="UserLogAppender"/> -->
  <!-- </Logger> -->

</Loggers>
</Configuration>

```

Unica Platform utilities

This section provides an overview of the Unica Platform utilities, including some details that apply to all the utilities and which are not included in the individual utility descriptions.

Location of utilities

Unica Platform utilities are located in the `tools/bin` directory under your Unica Platform installation.

List and descriptions of utilities

The Unica Platform provides the following utilities.

- [Clientdetails \(on page 246\)](#) - generates a key for a client application like Unica Journey to authenticate with one Unica Platform instance.
- [alertConfigTool \(on page 241\)](#) - registers alerts and configurations for Unica products
- [configTool \(on page 242\)](#) - imports, exports, and deletes configuration settings, including product registrations

- [datafilteringScriptTool \(on page 247\)](#) - creates data filters
- [encryptPasswords \(on page 248\)](#) - encrypts and stores passwords
- [encryptTomcatDBPasswords \(on page 249\)](#) - encrypt the database passwords that the Tomcat Application server uses internally
- [partitionTool \(on page 250\)](#) - creates database entries for partitions
- [populateDb \(on page 253\)](#) - populates the Unica Platform database
- [quartzjobtool \(on page 257\)](#) - Update scheduler jobs created in version 11.1 and older versions
- [restoreAccess \(on page 253\)](#) - restores a user with the platformAdminRole role
- [scheduler_console_client \(on page 255\)](#) - lists or starts Unica Scheduler jobs that are configured to listen for a trigger.
- `insightsdbutil` - Installer places report design files which possesses database connection tokens. You must update them for your system database. You must run `insightsdbutil.sh/bat` utility to update the same. See the Unica Insights Installation and Configuration Guide for more details.

Prerequisites for running Unica Platform utilities

The following are prerequisites for running all Unica Platform utilities.

- Run all utilities from the directory where they are located (by default, the `tools/bin` directory under your Unica Platform installation).
- On UNIX™, the best practice is to run the utilities with the same user account that runs the application server on which Unica Platform is deployed. If you run a utility with a different user account, adjust the permissions on the `platform.log` file to allow that user account to write to it. If you do not adjust permissions, the utility is not able to write to the log file and you might see some error messages, although the tool should still function correctly.

Authentication of utilities

Utilities such as `configTool` and other Unica back end utilities are designed to be used by system administrators and require physical access to the host servers for them to be invoked. For this reason, authentication for these utilities has been designed to be independent of the UI authentication mechanism. Access to these utilities is available to users with Unica Platform administrator privileges. Access to these utilities is expected to be locally defined in Unica Platform and authenticated against the same.

Troubleshooting connection issues

All of the Unica Platform utilities except `encryptPasswords` interact with the Unica 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 Unica Platform was installed. This information is stored in the `jdbc.properties` file, located in the `tools/bin` directory under your Unica 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 Unica Platform installation, or on the command line. The Unica 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 Unica Platform utilities

The following options are available in all Unica 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 Unica 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 Unica Platform utilities on additional machines

On the machine where the Unica Platform is installed, you can run the Unica Platform utilities without any additional configuration. However, you might want to run the utilities from another machine on the network. This procedure describes the steps required to do this.

Ensure that the machine on which you perform this procedure meets the following prerequisites.

- The correct JDBC driver must exist on the machine or be accessible from it.
- The machine must have network access to the Unica Platform system tables.
- The Java™ runtime environment must be installed on the machine or be accessible from it.

1. Gather the following information about the Unica 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 Unica 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 Unica installer and install the Unica Platform.

Enter the database connection information that you gathered for the Unica Platform system tables. If you are not familiar with the Unica installer, see the Unica Campaign or Unica Plan installation guide.

You do not have to deploy the Unica Platform web application if you are installing the utilities only.

Utilities

This section describes the Unica Platform utilities, with functional details, syntax, and examples.

alertConfigTool

Notification types are specific to the various Unica 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 Unica Platform installation.

```
alertConfigTool -i -f Platform_alerts_configuration.xml
```

configTool

The properties and values on the **Configuration** page are stored in the Unica Platform system tables. You can use the `configTool` utility to import and export configuration settings to and from the system tables.

When to use configTool

You might want to use `configTool` for the following reasons.

- To import partition and data source templates that are supplied with Unica Campaign, which you can then modify and duplicate by using the **Configuration** page.
- To register (import configuration properties for) Unica 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 Unica.
- 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 Unica 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 Unica 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 Unica 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 Unica 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 Unica 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 Unica Platform installer as described in the *Unica Platform Installation Guide*.
- After the initial installation, to re-register products other than the Unica 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 Unica, 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 74. Product names for configTool registration and unregistration

Product name	Name used in configTool
Unica Platform	Manager
Unica Campaign	Campaign
Unica Collaborate	Collaborate
Unica Deliver	Deliver
Unica Journey	Journey
Unica Insights	UnicaInsights
Unica Content Integration	assetPicker
Unica Offer	Offer
Unica Interact	interact
Unica Optimize	Optimize
Unica Plan	Plan
Opportunity Detect	Detect
IBM SPSS Modeler Advantage Enterprise Marketing Management Edition	SPSS
Digital Analytics	Coremetrics

-u *productName*

Unregister an application that is specified by *productName*. You do not have to include a path to the product category; the product name is sufficient, and it is required. The process removes all properties and configuration settings for the product.

Options

-o

When used with **-i** or **-r**, it overwrites an existing category or product registration (node).

When used with **-d**, you can delete a category (node) that does not have the **Delete Category** link on the **Configuration** page.

Examples

- Import configuration settings from a file named `Product_config.xml` in the `conf` directory under the Unica Platform installation.

```
configTool -i -p "Affinium" -f Product_config.xml
```

- Import one of the supplied Unica Campaign data source templates into the default Unica Campaign partition, `partition1`. The example assumes that you placed the Oracle data source template, `OracleTemplate.xml`, in the `tools/bin` directory under the Unica 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 Unica 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 Unica 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 Unica Platform installation, and force it to overwrite an existing registration of this application.

```
configTool -r product Name -f app_config.xml -o
```

- Unregister an application named `productName`.

```
configTool -u productName
```

- Run the following command to enable `encodeCSV` feature:

```
configTool -vp -p "Affinium|Plan|umoConfiguration" -f Plan_Home\conf\Plan_encodeProperty_11.1.xml
```

- Register Unica Interact Settings as configuration menu under `AffiniumWebApps\Campaign\interact\conf\interact_setup_navigation.xml` using

```
configTool.bat -v -i -p "Affinium|suite|uiNavigation|settingsMenu" -f "interact_setup_navigation.xml"
```

Clientdetails

This utility generates keys for client application, like Unica Journey to authenticate with one Platform instance.

It registers the key in Platform database and prints it on the console. The key can then be copied and pasted on the target application.

Syntax

```
clientDetails -a appName
```

Commands

```
-a appName
```

Generate the key for the specified application. Possible values for appName is Manager (for Unica Platform) and Journey (for Unica Journey)

Examples

Generate the key for Unica Journey

```
clientDetails -a Journey
```

datafilteringScriptTool

The `datafilteringScriptTool` utility reads an XML file to populate the data filtering tables in the Unica 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 Unica 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 Unica Platform must be deployed and running.

Using datafilteringScriptTool with SSL

When the Unica Platform is deployed using one-way SSL you must modify the `datafilteringScriptTool` script to add the SSL options that perform handshaking. To modify the script, you must have the following information.

- Truststore file name and path
- Truststore password

In a text editor, open the `datafilteringScriptTool` script (`.bat` or `.sh`) and find the lines that look like this (examples are Windows™ version).

```
:callexec
```

```
"%JAVA_HOME%\bin\java" -DUNICA_PLATFORM_HOME="%UNICA_PLATFORM_HOME%"
```

```
com.unica.management.client.datafiltering.tool.DataFilteringScriptTool %*
```

Edit these lines to look like this (new text is in **bold**). Substitute your truststore path and file name and truststore password for `myTrustStore.jks` and `myPassword`.

```
:callexec
```

```
SET SSL_OPTIONS=-Djavax.net.ssl.keyStoreType="JKS"
```

```
-Djavax.net.ssl.trustStore="C:\security\myTrustStore.jks"
```

```
-Djavax.net.ssl.trustStorePassword=myPassword
```

```
"%JAVA_HOME%\bin\java" -DUNICA_PLATFORM_HOME="%UNICA_PLATFORM_HOME%" %SSL_OPTIONS%
```

```
com.unica.management.client.datafiltering.tool.DataFilteringScriptTool %*
```

Syntax

```
datafilteringScriptTool -r pathfile
```

Commands

```
-r path_file
```

Import data filter specifications from a specified XML file. If the file is not located in the `tools/bin` directory under your installation, provide a path and enclose the `path_file` parameter in double quotation marks.

Example

- Use a file named `collaborateDataFilters.xml`, located in the `C:\unica\xml` directory, to populate the data filter system tables.

```
datafilteringScriptTool -r "C:\unica\xml\collaborateDataFilters.xml"
```

encryptPasswords

The `encryptPasswords` utility is used to encrypt and store either of two passwords that Unica Platform uses internally.

The two passwords that the utility can encrypt are as follows.

- The password that the Unica 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 Unica Platform installation) with a new one.
- The keystore password used by the Unica Platform when it is configured to use SSL with a certificate other than the default one supplied with the Unica 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 Unica 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 Unica Platform installation.
- Before running `encryptPasswords` to encrypt and store the keystore password, you must have created or obtained a digital certificate and know the keystore password.

Syntax

```
encryptPasswords -d databasePassword
```

```
encryptPasswords -k keystorePassword
```

Commands

-d databasePassword

Encrypt the database password.

-k keystorePassword

Encrypt the keystore password and store it in a file named `pfile`.

Examples

- When the Unica 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 Unica 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
```

encryptTomcatDBPasswords

The `encryptTomcatDBPasswords` utility is used to encrypt the database passwords that the Tomcat Application server uses internally. It is used to encrypt database passwords used in `Campaign.xml` and `unica.xml`. This utility can encrypt the Unica application database password. The utility prints the encrypted password in the command line.

When to use encryptTomcatDBPasswords

Use `encryptTomcatDBPasswords` utility when you want to use encrypted password under Tomcat configurations. It can be used when the Campaign or Unica System DB password has expired or changed. You can use this utility and encrypt the password and this will be get replaced in Campaign.xml, unica.xml and plan.xml located at `<instanceHome>\conf\Catalina\localhost.`

Syntax

```
encryptTomcatDBPasswords -d databasePassword
```

Commands

```
-d databasePassword
```

Encrypt the database password.



Note:

This utility is available only when the user selects Tomcat as the application server while installing Unica Platform.

This utility can be used only in case when the user wants use encrypted passwords instead of plain text passwords, under Tomcat configurations.

For more details, see the Tomcat documentation.

partitionTool

Partitions are associated with Unica Campaign policies and roles. These policies and roles and their partition associations are stored in the Unica Platform system tables. The `partitionTool` utility seeds the Unica 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 Unica Platform system tables with basic policy and role information.

See the installation guide appropriate for your version of Unica Campaign for detailed instructions on setting up multiple partitions in Unica Campaign.

Special characters and spaces

Any partition description or user, group, or partition name that contains spaces must be enclosed in double quotation marks.

Syntax

```
partitionTool -c -s sourcePartition -n newPartitionName [-u admin_user_name] [-d partitionDescription] [-g
groupName] [-a application]
```

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 Unica user with the Admin role in both the Administrative Roles policy and the global policy in Unica Campaign. The partition name you specify is automatically set as this user's password.
- Creates a new Unica 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 Unica 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.

`-a application`

Optional, used with `-c`, `-n`, `-g`, and `-u` only. Clones data from the source partition for the specified application only partition. The application must be Unica Suite application.

`-g groupName`

Optional, used with `-c` only. Specifies the name of the Unica Platform Admin group that the utility creates. The name must be unique within this instance of Unica 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 Unica 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"`
- Update a partition with the following characteristics.
 - Cloned from partition1
 - Partition name is `partition2`
 - Specify admin user name and admin user group of partition2
 - `partitionTool -c -s partition1 -n partition2 -u partition2AdminUser -a Journey`



Note: While using `-a` option ensure to specify the group name, if the group name was specified explicitly when partition was created by utility.



```
partitionTool -c -s partition1 -n partition2 -u partition2AdminUser -g [partition2 group name] -a
```

Journey

populateDb

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

The Unica installer can populate the Unica Platform system tables with default data for Unica Platform and for Unica 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 Unica Platform system tables, you must insert default data in the Unica Platform system tables using this utility.

For Unica Campaign, this data includes security roles and permissions for the default partition. For Unica 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 Unica Platform system tables. Valid product names are `Manager` (for Unica Platform) and `Campaign` (for Unica Campaign).

Examples

- Insert Unica Platform default data manually.

```
populateDb -n Manager
```

- Insert Unica Campaign default data manually.

```
populateDb -n Campaign
```

restoreAccess

The `restoreAccess` utility allows you to restore access to Unica Platform if all users with `PlatformAdminRole` privileges have been inadvertently locked out or if all ability to log in to the Unica 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 Unica 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 Unica 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 Unica Platform as an internal user, that user's password is changed.

Only a user with the login name of PlatformAdmin and with PlatformAdminRole privileges can universally administer all dashboards. So if the platform_admin user is disabled and you create a user with `restoreAccess`, you should create a user with a login of platform_admin.

Improper configuration of NTLMv2 authentication

If you implement NTLMv2 authentication with improper configuration and can no longer log in, use `restoreAccess` to restore the ability to log in.

When you run `restoreAccess` in this way, the utility changes the value of the `Platform | Security | Login method` property to `Unica 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 Unica 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 `Unica 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 `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 Unica 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 Unica 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 Unica 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 Unica Platform installation or on the command line.
- Import the SSL certificate used by the web application server on which the Unica 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.



Note: In case of upgrades, JRE shipped with Unica is overwritten so ensure that you reimport certificates in the JRE if you are using the same JRE.

- 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 Unica 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 Unica 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 Unica Platform must be installed, deployed, and running.

Syntax

```
scheduler_console_client -v -t trigger_name user_name
```

```
scheduler_console_client -s -t trigger_name user_name
```

Commands

`-v`

List the scheduler jobs configured to listen for the specified trigger.

Must be used with the `-t` option.

`-s`

Send the specified trigger.

Must be used with the `-t` option.

Options

`-t trigger_name`

The name of the trigger, as configured in the scheduler.

Example

- List jobs configured to listen for a trigger named `trigger1`.

```
scheduler_console_client -v -t trigger1 myLogin
```

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

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

quartzjobtool

Scheduler jobs created in version 11.1 or older versions require to be updated to run on version 12.0. Use the `quartzjobtool` utility to update the scheduler jobs when the installer has not done this automatically during installation or upgrade. This tool reads environment variables from the `setenv_quartz` script. The Unica Platform installer should have set this variable automatically, 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).

Syntax

`quartzjobtool`

Use the `quartzjobtool` to update scheduler jobs. This is a required step. If this upgrade tool is not run, any existing scheduled job will fail to start. The `quartzjobtool` is in the `tools\bin` directory under Unica Platform installation. Run this utility from the `tools\bin` directory.

Example command (Windows): `quartzjobtool.bat`

Example command (Unix): `./quartzjobtool.sh`

Example

Update scheduler jobs `quartzjobtool`

Unica Platform SQL scripts

This section describes the SQL scripts provided with Unica Platform to perform various tasks relating to the Unica Platform system tables.

The Unica Platform SQL scripts are located in the `db` directory under your Unica Platform installation.

The scripts are designed to be run against the Unica Platform system tables, using the database client.

ManagerSchema_DeleteAll.sql

The `Manager_Schema_DeleteAll.sql` script removes all data from the Unica Platform system tables without removing the tables themselves. This script removes all users, groups, security credentials, data filters, and configuration settings from Unica 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 Unica Platform.

Additional requirements

To make Unica Platform operational after running `ManagerSchema_DeleteAll.sql`, you must perform the following steps.

- Run the `populateDB` utility. The `populateDB` utility restores the default configuration properties, users, roles, and groups, but does not restore any users, roles, and groups you have created or imported after initial installation.
- Use the `configTool` utility with the `config_navigation.xml` file to import menu items.
- If you have performed any post-installation configuration, such as creating data filters or integrating with an LDAP server or web access control platform, you must perform these configurations again.
- If you want to restore previously existing data filters, run the `datafilteringScriptTool` utility using the XML originally created to specify the data filters.

ManagerSchema_PurgeDataFiltering.sql

The `ManagerSchema_PurgeDataFiltering.sql` script removes all data filtering data from the Unica 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 Unica 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 Unica 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 Unica Platform system tables from a database. This script removes all tables, users, groups, security credentials, and configuration settings from Unica Platform.



Note: If you run this script against a database containing an earlier version of the Unica 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 Unica Platform where the system tables are in a database that contains other tables you want to continue using.

Additional requirements

To make the Unica Platform operational after running this script, you must perform the following steps.

- Run the appropriate SQL script to re-create the system tables.
- Run the `populateDB` utility. Running the `populateDB` utility restores the default configuration properties, users, roles, and groups, but does not restore any users, roles, and groups you have created or imported after initial installation.
- Use the `configTool` utility with the `config_navigation.xml` file to import menu items.
- If you have performed any post-installation configuration, such as creating data filters or integrating with an LDAP server or web access control platform, you must perform these configurations again.

SQL scripts for creating system tables

Use the scripts described in the following table to create Unica Platform system tables manually when your company policy does not allow you to use the installer to create them automatically.

The scripts are shown in the order in which you must run them.

Table 75. Scripts for creating system tables

Datasource Type	Script Names
IBM® DB2®	<ul style="list-style-type: none"> • <code>ManagerSchema_DB2.sql</code> <p>If you plan to support multi-byte characters (for example, Chinese, Japanese, or Korean), use the <code>ManagerSchema_DB2_unicode.sql</code> script.</p>

Table 75. Scripts for creating system tables (continued)

Datasource Type	Script Names
	<ul style="list-style-type: none"> • ManagerSchema_DB2_CeateFKConstraints.sql • active_portlets.sql • notification_rules.sql
Microsoft™ SQL Server	<ul style="list-style-type: none"> • ManagerSchema_SqlServer.sql • ManagerSchema__SqlServer_CeateFKConstraints.sql • active_portlets.sql • notification_rules.sql
MariaDB	<ul style="list-style-type: none"> • ManagerSchema_MariaDB.sql • ManagerSchema_MariaDB_StoredProcedures.sql • ManagerSchema_MariaDB_CreateFKConstraints.sql • active_portlets.sql • notification_rules.sql
Oracle	<ul style="list-style-type: none"> • ManagerSchema_Oracle.sql • ManagerSchema__Oracle_CeateFKConstraints.sql • active_portlets.sql • notification_rules_Oracle.sql
PostgreSQL	<ul style="list-style-type: none"> • ManagerSchema_PostgreSql.sql • ManagerSchema_PostgreSql_CreateFKConstraints.sql • ManagerSchema_pgsql_upg.sql • ManagerSchema_PostgreSQL_StoredProcedures.sql • ContentIntegration_pgsql.sql • active_portlets_PostgreSQL.sql

If you plan to use the scheduler feature that enables you to configure a flowchart to run at predefined intervals, you must also create the tables that support this feature. To create the scheduler tables, run the appropriate script, as described in the following table.

Table 76. Scripts for enabling the Unica Scheduler

Data Source Type	Script Name
DB2®	quartz_db2.sql

Table 76. Scripts for enabling the Unica Scheduler (continued)

Data Source Type	Script Name
Microsoft™ SQL Server	quartz_sqlServer.sql
Oracle	quartz_oracle.sql
MariaDB	quartz_MariaDB.sql
PostgreSQL	quartz_PostgreSql.sql

When to use the create system tables scripts

You must use these scripts when you install or upgrade Unica 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 Unica Platform system tables from your database.

Unica configuration properties

This section describes the configuration properties found on the **Settings & Configuration** page.

Unica Platform configuration properties

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

Unica Platform

Properties in this category allow you to set the default locale, and to set flags for whether the installation of Unica Platform is clustered, whether Unica Plan is integrated with Unica Campaign, and whether offer integration is enabled for the integration.

Region

Description

Specifies the locale preference for Unica users. When you set this property on the Configuration page, the setting you apply is the default setting throughout Unica for all users, except those whose locale preference is set individually through the Unica Platform User page. When you set this property for an individual user, the setting you apply for that user overrides the default setting.

This preference setting affects display of the language, time, numbers, and dates in Unica applications.

Availability of locales may vary depending on the Unica application, and not all applications support this locale setting in the Unica 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 hosted online help is installed. If Unica users have internet access, you should not change the default value, which points to the online help server maintained and updated by .

Default value

The URL of the hosted help server.

Valid Values

Any server on which hosted help is installed.

Unica Plan - Unica Campaign integration

Description

A flag indicating whether Unica Plan and Unica Campaign are installed together and integrated. For more information about configuring this integration, see the Unica Plan and Unica Campaign Integration Guide.

Default value

False

Valid Values

True | False

Unica Plan - Offer integration

Description

For systems that integrate Unica Plan with Unica Campaign, this flag indicates whether offer integration is also enabled. Offer integration enables the ability to use Unica Plan to perform offer lifecycle management tasks. For more information about configuring this integration, see the Unica Plan and Unica 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 Unica. The default is the default dashboard.

Default value

The default dashboard.

Valid Values

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

Domain name**Description**

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

If users access Unica products with the Chrome browser, use the fully qualified domain name (FQDN). If the FQDN is not used, the Chrome browser cannot access the product URLs.

Default value

Not defined

Disable page tagging**Description**

When set to the default value of `False`, uses the Site ID code that was entered during Unica Platform installation to gather basic statistics that track overall product usage trends to develop and improve products. sends the information to <http://pt200201.unica.com> over HTTP.

If you do not want to have such information collected, set this property to `True`.

Default value

`False`

Valid values

`True` | `False`

Is this deployment clustered**Description**

If you install Unica Platform in a clustered deployment, set this property to `True`. Otherwise, retain the default value of `False`.

If you change this property while Unica Platform is running, you must restart Unica Platform for the changes to take effect.

Default value

`False`

Valid values

`True` | `False`

Apply security on static content for all applications

Description

When this value is set to `yes`, if an authenticated user attempts to directly access any static content such as an image, a check is performed to verify the user's authentication. If the user is authenticated, the content is rendered. If the user is not authenticated, the user is sent to the login page. This setting applies across all Unica products.

Default value

No

Valid values

Yes | No

Unica | General | Navigation

Properties in this category specify values that are used internally to navigate among Unica products.

TCP port for secure connections

Description

Specifies the SSL port in the web application server on which the Unica Platform is deployed. This property is used internally for communication among Unica products.

Default value

7001

TCP port for standard connections

Description

Specifies the HTTP port in the web application server on which the Unica Platform is deployed. This property is used internally for communication among Unica products.

Default value

7001

Unica Platform URL

Description

Specifies the URL used for Unica Platform. This is set at installation time and normally should not be changed. Note that the URL contains the domain name, as shown in the following example.

```
protocol://machine_name_or_IP_address.domain_name:port_number/context-root
```

The machine name should not be `localhost`.

If users access Unica products with the Chrome browser, use the fully qualified domain name (FQDN) in the URL. If the FQDN is not used, the Chrome browser cannot access the product URLs.

! **Important:** If Unica products are installed in a distributed environment, you must use the machine name rather than an IP address in the navigation URL for all of the applications in the suite. Also, if you are on a clustered environment and choose to use ports that are different from the default ports 80 or 443 for your deployment, do not use a port number in the value of this property.

Default value

Not defined

Example

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

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

Unica Platform Internal URL

Description

Specifies the internal URL used for Unica Platform. This needs to be set explicitly as per internal platform server configuration.

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 Unica products are installed in a distributed environment, you must use the machine name rather than an IP address in the navigation URL for all the applications in the suite. Also, if you are on a clustered environment and choose to use ports that are different from the default ports 80 or 443 for your deployment, do not use a port number in the value of this property.

Default value

Not defined

Example

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

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

Unica | General | Data filtering

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

Default table name

Description

This configuration property is required to enable data filters.

Set the value of this property to exactly match the name used for the `addTables` | `AddDataTable` | `dataTable` | `name` element in the XML used to create the data filters.

Default value

Undefined

Valid values

Maximum of 50 characters of type varchar.

Default audience name

Description

This configuration property is required to enable data filters.

Set the value of this property to exactly match the name used for the `AddAudience` | `audience` | `name` element in the XML used to create the data filters.

Default value

Undefined

Valid values

Maximum of 50 characters of type varchar.

Enable data filter cache

Description

This property is optional, and can be set to improve data filter performance.

This property specifies whether the Unica Platform retrieves data filter definitions from the database or from a cache. When this value is **true**, data filter definitions are stored in the cache and the cache is updated whenever there is any change in the data filter definitions.

You must restart the Unica Platform web application after you make a change in this property value before it can take effect.

Default value

False

Unica | General | Password settings

Properties in **General | Password Settings** category specify the policies that apply to Unica passwords. Most of these password options apply only to passwords for internal users (created within the Unica 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 Unica system, and no one can log in as that user.

If set to zero or less, the system allows an infinite number of consecutive failures.

Default value

3

Valid values

Any integer

Password history count

Description

Specifies the number of old passwords the system retains for a user. The user is not allowed to reuse any passwords within this list of old passwords. If the value is set to zero or less, then no history is retained, and the user may reuse the same password repeatedly. Note that the password history count does not include the password initially assigned to a user account when it is created.

Default value

0

Valid values

Any integer

Validity (in days)

Description

Specifies the number of days before a user's password expires.

If the value is zero or less, then the password never expires.

If the value is greater than zero, users are required to change their password the first time they log in, and the expiration interval is counted from the date of the first login.

If you change this value after users and passwords have been created, the new expiration date takes effect for existing users the next time they change their password.

Default value

30

Valid values

Any integer

Blank passwords allowed

Description

Specifies whether the a blank password is allowed.If you set this to true you should also set `Minimum character length=0`.

Default value

true

Valid values

true | false

Allow identical user name and password

Description

Specifies whether the user's password is allowed to be the same as the user's login name.

Default value

false

Valid values

true | false

Minimum number of numeric characters

Description

Specifies the minimum number of numbers required in a password. If the value is zero or less, then there is no minimum requirement.

Default value

0

Valid values

Any integer

Minimum number of letter characters

Description

Specifies the minimum number of letters required in a password. If the value is zero or less, then there is no minimum requirement.

Default value

0

Valid values

Any integer

Minimum character length

Description

Specifies the minimum length of a password. If the value is zero or less, then there is no minimum requirement. If you set the value to greater than 0, you should also set `Blank passwords allowed=false`.

Default value

4

Valid values

Any integer

Minimum number of special characters

Description

Specifies the minimum number of special characters required in a password. This is applicable for values more than zero. You can only use the following special characters when you create a password.

- Asterisk "*"
- Exclamation "!"
- The at sign "@"
- Dollar "\$"
- Ampersand "&"
- Hash "#"

Default value

0

Valid values

Any integer

Minimum number of lowercase characters

Description

Specifies the minimum number of lowercase letters required in a password. It is applicable to values greater than 0, where you must set the corresponding value for `Minimum number of letter characters`.

Default value

0

Valid values

Any integer

Minimum number of uppercase characters

Description

Specifies the minimum number of uppercase letters required in a password. It is applicable to values greater than 0, where you must set the corresponding value for `Minimum number of letter characters`.

Default value

0

Valid values

Any integer

Maximum character length

Description

Specifies the maximum length of a password. It is applicable to values greater than 0.

Default value

0

Valid values

Any integer

Unica | General | Miscellaneous

Properties in this category specify values that are used internally, as well as a value you may need to set for the locale.

Token lifetime

Description

Specifies the length of time, in seconds, that a token generated by the Unica Platform is valid. It is part of the suite sign-on implementation, and you should not change this value.

Default value

15

Valid values

Any positive integer

Default language**Description**

Specifies the default language for the Unica Platform. If you plan to install Unica Campaign, you should set this value to match the locale set for Unica Campaign in the `defaultLocale` property for Unica Campaign.

Default value

English

Valid values

Supported locales

Unica | General | Communication | Email

Properties in this category are used to configure the Unica Platform to send emails to users for system alerts and notifications.

Enable email communication**Description**

When set to `True`, the Unica Platform attempts to send emails to users for system alerts and notifications. The other properties in this category must also be set to enable this feature.

Default value

False

Email server protocol**Description**

Specifies the protocol on the mail server that is used for sending system alerts and notifications to users. This is required for email notifications.

Default value

smtp

Email server host**Description**

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

Default value

localhost

Email server port

Description

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

Default value

25

'From' address for emails

Description

Specifies the account from which system alert and notification emails are sent. If authentication is required on your mail server, use the email address of the account that you used when you saved a mail server account name and password as a data source in a Unica Platform user account. This is required for email notifications.

Default value

Not defined

Authentication required for mail server?

Description

Specifies whether the mail server requires authentication.

Default value

False

Unica user for email account

Description

Specifies the user name of the Unica Platform account where the email credentials are stored as a data source.

Required for notifications, only if your mail server requires authentication.

Default value

asm_admin

Data source for email account

Description

Specifies the name of the data source in the Unica Platform account where the email credentials are stored.

Required for notifications, only if your mail server requires authentication.

Default value

emailDS

Unica Platform | Scheduler

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

Client polling interval (ms)

Configuration category

Platform | Scheduler

Description

Unica Campaign polls the Unica Scheduler for jobs at regular intervals, specified in milliseconds by this value. The default value is 60 seconds. Avoid setting this property to any value less than 10000 (10 seconds), because doing so can decrease campaign performance.

Default value

60000

Client initialization delay (ms)

Description

The amount of time, expressed in milliseconds, that the Unica Campaign scheduler thread waits before polling the Unica Scheduler for jobs when Unica Campaign first starts up. Set this value to be at least as long as it takes for Unica Campaign to fully start up on your system. The default value is five minutes.

Default value

300000

Valid Values

Any integer

Maximum unknown status polling count

Description

Specifies the number of times the scheduler checks the status of a scheduled run whose status cannot be determined. After this limit is reached, the run status is listed as Unknown on the **Settings > Schedule management** page.

Default value

5

Valid Values

Any integer

Enable Scheduler

Description

Specifies whether the scheduler is enabled. Set this property to False if you do not want users to be able to use the scheduler. The False setting turns off the scheduler for all products that use it.

You must restart the Unica Platform web application when you enable or disable the scheduler.

Default value

True

Valid Values

True | False

Unica Platform | Scheduler | Recurrence definitions

Properties in this category set the recurrence patterns for the Unica 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 *

Unica Platform | Scheduler | Schedule registrations | [Product] | [Object type]

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

Executor class name

Description

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

Default value

Status polling interval

Configuration category

`Platform|Scheduler|Schedule registrations|[Product]|[Object type]`

For Unica Campaign flowcharts, the path for this property is `Platform|Scheduler|Schedule registrations|Campaign|Flowchart`

Description

The Unica Scheduler polls the product at regular intervals to obtain the run status of scheduled objects (for example, flowcharts or mailings) that have not reported a status. The interval is specified in milliseconds. The default value is 10 minutes. A more frequent polling interval (a smaller value) can negatively affect system performance. A less frequent polling interval (a larger value) reduces the load

on the system. For Unica Campaign, set a less frequent polling interval when you have a large number of Unica Campaign flowcharts that take more than 10 minutes to complete.

Default value

600000

Name of group to receive job notifications

Description

Notifications for all schedules for each object type are sent to all members of the group you specify here.

Unica 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 Unica scheduler. Note that these default groups do not appear on the User groups page. You can use the throttling group template to create additional groups.

Throttling threshold

Description

The greatest number of schedules associated with this group that can run concurrently. The groups you specify here appear in the **Scheduler group** drop-down list in the scheduler user interface for creating and editing schedules. The default throttling group is set to 999, which is effectively no limit. Because all schedules must belong to a throttling group, you should leave this value unchanged so that schedules that you do not want to throttle can be assigned to this group.

Default value

Valid Values

Any positive integer.

Unica Platform | Security

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

Login method

Description

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

- If you set the value to `Unica Platform`, Unica products use the Unica Platform for authentication and authorization.
- If you set the value to `LDAP`, Unica products use an LDAP server for authentication.
- If you set the value to `Web access control`, Unica products use web access control software for authentication.
- If you set the value to `SAML 2.0`, Unica products use your IdP server for authentication.

If you change this setting, stop and restart the Unica Platform web application so that your change takes effect.

Default value

`Unica Platform`

Valid Values

`Unica Platform | LDAP | Web access control`

Unica 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 Unica 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 Unica 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 Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

Use credentials stored in Unica Platform

Description

Specifies whether the Unica Platform uses credentials from the Unica Platform database when searching the LDAP or Windows™ Active Directory server during user authentication (at login time).

If this value is `true`, the Unica Platform uses credentials from the Unica Platform database, and you must set the appropriate values for the `Unica 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 Unica 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 Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

Unica Platform user for LDAP credentials

Description

Specifies the name of the Unica user that has been given LDAP administrator login credentials. Set this value if you set the `Use credentials stored in UUnica Platform` property in this category to `true`.

Set the value of this property to the user name you created for the Unica 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 Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

Data source for LDAP credentials

Description

Specifies the Unica Platform data source for LDAP administrator credentials. Set this value if you set the `Use credentials stored in Unica Platform` property in this category to `true`.

Set the value of this property to the data source name you created for the Unica user when you configured LDAP integration. This property works in conjunction with the `Unica Platform user for LDAP credentials` property in this category.

Default value

Undefined

Availability

This property is used only when the Unica 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 Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

Require SSL for LDAP connection

Path

Unica Platform | Security | LDAP

Description

Specifies whether the Unica 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 IBM Security Access Manager is `\w*`

You should also use this value when you use a custom proxy to integrate Unica Campaign hosted on premises and Digital Analytics in the cloud.

Default value

Undefined

Valid Values

Any Java™ regular expression.

Availability

This property is used only when the Unica Platform is configured to integrate with web access control software.

Web access control header variable

Description

Specifies the HTTP header variable configured in the web access control software, which is submitted to the web application server. By default, SiteMinder uses `sm_user` and IBM Security Access Manager (SAM) uses `iv-user`. For SAM, set this value to the user name component of the Raw string, not the HTTP string.

Default value

Undefined

Valid Values

Any string

Availability

This property is used only when the Unica Platform is configured to integrate with web access control software.

Additional header variables

Description

Specifies comma separated list of additional HTTP header variable to be captured while logging through web access control software. Specified HTTP header variables are captured and stored in authentication audit log if audit logs are enabled.

Default value

Undefined

Valid Values

Any comma separated string

Availability

This property is used only when the Unica Platform is configured to integrate with web access control software.

Platform | Security | Login method details | SAML 2.0

Properties in this category configure single sign-on through a SAML 2.0 IdP server.

IdP server URL for single sign-in

Description

The URL of the page that appears when users open the single sign-on URL to Unica.

Default value

[CHANGE ME]

IdP server URL for single sign-out**Description**

Optional. When users log out, they can be redirected to the page you set here so that their logout also logs them out of the IdP server. Your IdP server is likely to provide a URL for this purpose.

Default value

[CHANGE ME]

Error page URL for SSO error**Description**

If an error occurs during single sign-on due to a configuration or integration issue, users can be redirected to the page specified here. This setting overrides the default error page provided by Unica Platform.

Default value

[CHANGE ME]

Destination URL**Description**

The URL of the service provider (application) to which the user is redirected after successful authentication through the IdP server. This URL appears in every SAML request under the <AuthnRequest Destination> tag.

Default value

[CHANGE ME]

Consumer service URL**Description**

The assertion consumer service URL that the service provider (application) consumes and parses for SAML assertions. This URL appears in every SAML request under the <AuthnRequest AssertionConsumerServiceURL> tag. This value can be the same as the value of the **Destination URL** property.

Default value

[CHANGE ME]

Application ID**Description**

The application ID assigned to Unica Platform in the IdP server. This ID is included in every SAML request to the IdP server. This ID appears in every SAML request under the <Issuer> tag.

Default value

[CHANGE ME]

Service provider name qualifier

Description

The service provider's name qualifier. This name qualifier appears in every SAML request under the <NameIDPolicy SPNameQualifier> tag.

Default value

[CHANGE ME]

Metadata path

Description

The location of the IDP metadata file on the Unica Platform server. This IDP metadata file is provided by the IDP server.

Default value

[CHANGE ME]

Entity ID

Description

The entity ID of the IdP server. Set this property to the value of *entityID* in the XML declaration at the top of the metadata file produced by the IdP server.

Unica Platform uses this ID during assertion validation to load the IdP configurations and digital certificate.

Default value

[CHANGE ME]

Attributes NVP for response parsing

Description

User account attributes are sent to Unica Platform by the IdP server. You can use this configuration property to capture attributes for users created in Unica Platform automatically, when the **Add authenticated users to Platform** property is enabled.

The IdP server might use a different name for an attribute compared to the name that Unica Platform uses. You can use this property to map the IdP attribute to the corresponding attribute in Unica Platform. This eliminates the need for code changes.

For example, the IdP server might use **emailAddress** as the name for an attribute that is named **Email** in Unica Platform. You would enter **Email=emailAddress** as a value in this property to map the attribute.

Use the following values for the user attributes in Unica Platform.

- FirstName
- LastName
- Department
- Organization
- Country
- Email
- Address1
- Address2
- Phone1

Use for work phone.

- Phone2

Use for mobile phone.

- Phone3

Use for home phone.

- AltLogin
- ExternalUsersGroup

If you enable the **Add authenticated users to Unica Platform** property, a user authenticated from the IdP server is created in Unica Platform if that user does not already have a Unica Platform account. These users are automatically added to a default user group, **ExternalUsersGroup**. However, you can also specify a custom group to which users are added. If you implement this option, set the value of the **ExternalUsersGroup** attribute to the name of the custom user group. For example, if you want a user to be added to a group name identified by the SAML attribute **MyGroup**, you would set this value to **ExternalUsersGroup=MyGroup**. Users will be added to the group name which is specified to **MyGroup** SAML attribute.

Separate multiple name-value pairs with a semi-colon.

Default value

```
omit-xml-declaration=yes;
```

Process encrypted IdP response

Description

If your IdP server is configured to send encrypted responses, enable this property to indicate that the SAML response from the IdP server must be decrypted using the configured shared key before Unica Platform processes it.

If you enable this property, you must also set the value of **Shared secret key** to the secret key that is used to decrypt the response.

Default value

Disabled

Shared secret key

Description

When the **Process encrypted IdP response** option is enabled, set this property value to the path of the keystore file.

Default value

[CHANGE ME]

Key store credential holder

Description

Set this value to the login name of the Unica user account that holds the SAML shared secret in a data source.

Default value

[CHANGE ME]

Key store credential data source

Description

Set this value to the name of the data source created to hold the shared secret used for decryption. The password in the data source is the password for the key store file.

Default value

[CHANGE ME]

Certificate alias

Description

When the **Process encrypted IdP response** option is enabled, set this property value to the certificate alias of the private key stored in the keystore file. This is used in decrypting the encrypted SAML response sent by the IDP server.

Default value

[CHANGE ME]

Add authenticated users to Platform

Description

When this option is enabled, a user authenticated from the IdP server is created in Unica Platform if that user does not already have a Unica Platform account.

Newly created users are automatically added to a default group, **ExternalUsersGroup**.

The **ExternalUsersGroup** has only the Unica Platform **UserRole**. An administrator must grant additional permissions for the newly created users to access and use Unica products. An administrator can grant additional permissions by making users members of groups with different application access levels.

Alternatively, the SAML response can contain a custom user group name, and newly created users are added to this group.

When this option is disabled, a user authenticated from the IdP server can not access Unica Platform, if that user does not have an account in Unica Platform.

Default value

Disabled

Redirect to SSO

Description

When this value is **True**:

- Users who log in to Unica are redirected to the IdP single sign on page
- After users log in, they go to the standard Unica Platform landing page.
- The standard Unica Platform login screen is never available.



Note:

- **Setting nameID format**

By default, SAML request is generated with nameID format as a transient. If you require to construct SAML request with persistent nameID format, you must set this JVM parameter.

```
-DENABLE_PERSISTENT_NAMEID_FORMAT=true
```

- **Configuring SAML authenticated user creation for non-default partition**

If you enable the `Add authenticated users to Unica Platform` property, a user authenticated from the IdP server is created in Unica Platform if that user do not have an Unica Platform account already. These users are automatically added under default partition, i.e., under partition with ID 1.

However, if you want a user to be added under a different partition then this must be specified as SAML attribute



Example: `PartitionId=<partitionid>`

- **Configuring RequestedAuthnContext in SAML request**

By default SAML request is generated with `RequestedAuthnContext` in the SAML request.

Some IDP server servers do not require `RequestedAuthnContext` in SAML request. In order to remove this from request, you must set this JVM parameter.

```
-REMOVE_REQUESTED_AUTHN_CONTEXT=true
```

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 Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

LDAP sync interval

Description

The Unica Platform synchronizes with the LDAP or Active Directory server at regular intervals, specified in seconds here. If the value is zero or less, the Unica 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 Unica 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 Unica Platform is started. For example an `LDAP sync delay` of 23:00 and an `LDAP sync interval` of 600 mean that when the Unica 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 Unica 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 Unica 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 Unica 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 Unica 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 Unica 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 Unica 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 Unica 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 Unica 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 Unica 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 Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

LDAP config Unica Platform group delimiter

Description

In the LDAP reference to Unica Platform group map category, if you want to map one LDAP or Active Directory group to multiple Unica 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 Unica 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 Unica Platform user creation category).

`FILTER` is optional: if omitted, the Unica 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 Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

Unica Platform user for LDAP credentials

Description

Specifies the name of Unica user that has been given LDAP administrator login credentials.

Set the value of this property to the user name you created for the Unica 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 Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

Data source for LDAP credentials

Description

Specifies the Unica Platform data source for LDAP administrator credentials.

Set the value of this property to the data source name you created for the Unica user when you configured LDAP integration. This property works in conjunction with the `Unica Platform user for LDAP credentials` property in this category.

Default value

Undefined

Availability

This property is used only when the Unica 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 Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

LDAP BaseDN periodic search disabled**Description**

When this property is set to `True`, the Unica Platform performs the LDAP synchronization search using the distinguished name set in the `Base DN` property under the **Unica Platform | Security | LDAP** category. If this property is set to `False`, the Unica Platform performs the LDAP synchronization search using the groups mapped to LDAP groups under **LDAP reference to Unica Platform group map**.

The following table describes whether changes are picked up in periodic synchronization, depending on the value set for this property.

Table 77. 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 Unica Platform, delete a user synchronized from the LDAP server	Yes	No
Remove a user from an LDAP group mapped to a Unica Platform group	No	No
In Unica Platform, remove a user from a Unica Platform group mapped to an LDAP group.	No	No
Add a new user to the LDAP server The change is picked up only when the login method is set to LDAP. If the login method is LDAP, the system will import new users from LDAP through auto sync.	Yes	Yes
Add a user to an LDAP group mapped to a Unica Platform group	Yes	No
Change user attributes on the LDAP server	Yes	Yes

Default value

False

Availability

This property is used only when the Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

User login

Description

Maps the Unica 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 Unica 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 Unica Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

`givenName`

Availability

This property is used only when the Unica 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 Unica Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

`sn`

Availability

This property is used only when the Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

User title

Description

Maps the Title user attribute in the Unica Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

title

Availability

This property is used only when the Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

Department

Description

Maps the Department user attribute in the Unica Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

Undefined

Availability

This property is used only when the Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

Company

Description

Maps the Company user attribute in the Unica Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

Undefined

Availability

This property is used only when the Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

Country

Description

Maps the Country user attribute in the Unica Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

Undefined

Availability

This property is used only when the Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

User email

Description

Maps the Email Address attribute in the Unica Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

mail

Availability

This property is used only when the Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

Address 1

Description

Maps the Address user attribute in the Unica Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

Undefined

Availability

This property is used only when the Unica 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 Unica Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

telephoneNumber

Availability

This property is used only when the Unica 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 Unica Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

Undefined

Availability

This property is used only when the Unica 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 Unica Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

Undefined

Availability

This property is used only when the Unica 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 Unica Platform to the equivalent user attribute in your LDAP or Active Directory server.

Default value

Undefined

Availability

This property is used only when the Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

Platform | Security | LDAP synchronization | LDAP reference to Unica 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 Unica Platform group specified in the `Unica 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 Unica 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 Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

Unica 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 Unica Platform group specified here.

Default value

Undefined

Availability

This property is used only when the Unica Platform is configured to integrate with a Windows™ Active Directory or other LDAP server.

Platform | Security | Federated authentication

The properties in this category are used in implementing SAML (Security Assertion Markup Language) 2.0 based federated authentication, which enables single sign-on among diverse applications.

Allow federated login

Description

Select the check box in this property to enable federated authentication in an integrated environment.

Default value

Disabled

Identity provider URL

Description

The URL of the identity provider server.

Certificate issuer

Description

The URL of the Certificate Authority that issued the certificate on the identity provider server. If you generate your own certificates using the Java™ keytool utility, set this value to the IdP server URL.

Platform | Security | Federated authentication | partitions | partition[n]

The properties in this category are used in implementing SAML (Security Assertion Markup Language) 2.0 based federated authentication between Unica applications and other and third party applications.

Keystore path

Description

The location of the trusted keystore file in the web application server.

Keystore passkey

Description

The passkey for the keystore in the web application server.

Keystore alias

Description

The alias for the keystore in the web application server.

Unica Platform | Security | API management

Properties in this category configure authentication behavior that applies to all Unica APIs.

Enable session-based API authentication

Description

If you select the check box for this property to enable it, users who are authenticated by logging in to Unica are not asked to log in again when they access a secure API from an Unica application during the session for which they are authenticated.

For example, when this property is enabled, and an authenticated Unica Interact user calls a Unica Campaign API during their session, no further login is required.

Default value

Disabled

Delete security token after a single use

Description

If you select the check box for this property to enable it, the token generated for an authenticated user is destroyed the first time this token is used for accessing any secure API. This enhances security by preventing any further use of the token.

Default value

Enabled

Platform | Security | API management | [Product] | (API configuration template)

Use the templates in this category to configure authentication for Unica APIs. You can block access, require HTTPS, or require authentication for APIs.

API URI

Description

For each product, the first part of the URI is resolved by the security framework, as follows:

```
http[s]://host:port/context root/api/product
```

Therefore, in this field you should enter only the resource name or names of the API you want to configure. You can obtain the string you need to enter from the product's API documentation.

The value used for this property must start with a / (forward slash); otherwise the configuration is ignored by the security framework.

This property supports an exact URL match as well a pattern match for the configured APIs.

- For an exact match, the URI may end with a forward slash (/) or the resource name.
- For a pattern match, the URI must end with an asterisk (*).

If you set the value of this property to `/*` the settings you use for the other properties in the category apply to all APIs for the product.



Note: For the Unica Platform `login` API, this configuration property is read-only.

Default value

Undefined

Block API access

Description

Select this option when you want to prevent an API from accessing a product. This option is not selected by default.

When an API is blocked, the security filter returns the HTTP status code 403 (forbidden).

Secure API access over HTTPS

Description

Select this option when you want to allow the API to access a product only over HTTPS. This option is selected by default.

When an API with this property enabled is accessed over HTTP rather than HTTPS, the security filter returns the HTTP status code 403 (forbidden).

Require authentication for API access

Description

Select this option when you want to require an API to authenticate before it can access a product. This option is selected by default.

When an API with this property enabled is accessed with invalid credentials, the security filter returns the HTTP status code 401 (unauthorized).



Note: For the Unica Platform `login` API, this configuration property is disabled, as this API is the first to be called for API authentication.

Default value

(Disabled)

Authentication mode

Description

Select this option when you want to authenticate API with Basic authentication or Bearer token authentication. When basic authentication or bearer token is selected, the corresponding user id and password has to be maintained in user's data source. For Manager authentication mode, it behaves in same way using parameter `api_auth_mode = Manager` in request header. This drop down selection is valid only if 'Require authentication for API access' is selected.

Default value

Manager

Data source credential holder

Description

Specify user name, which contains data source with required authentication credentials. Data source contains user id and password in case Basic authentication is selected in the Authentication mode drop down list. Data source contains bearer token in case Bearer token is selected in the Authentication mode drop down list.

Default value

asm_admin

Data source

Description

Specify data source name which is created under user specified in 'Data source credential holder'.

Default value

API_SECRET_DS

Unica Platform | Security | API management | [Product] | Unica Platform | Authentication

(Affinium|suite|security|apiSecurity|manager|managerAuthentication)Use the templates in this category to configure authentication for Unica APIs. You can block access, require HTTPS, or require authentication for APIs.

API URI

Description

For each product, the first part of the URI is resolved by the security framework, as follows:

`http[s]://host:port/context root/api/product`

Therefore, in this field you should enter only the resource name or names of the API you want to configure. You can obtain the string you need to enter from the product's API documentation.

The value used for this property must start with a / (forward slash); otherwise the configuration is ignored by the security framework.

This property supports an exact URL match as well a pattern match for the configured APIs.

- For an exact match, the URI may end with a forward slash (/) or the resource name.
- For a pattern match, the URI must end with an asterisk (*).

If you set the value of this property to `/*` the settings you use for the other properties in the category apply to all APIs for the product.



Note: For the Unica Platform `login` API, this configuration property is read-only.

Default value

`/authentication/login`

Block API access

Description

Select this option when you want to prevent an API from accessing a product. This option is not selected by default.

When an API is blocked, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Secure API access over HTTPS

Description

Select this option when you want to allow the API to access a product only over HTTPS. This option is selected by default.

When an API with this property enabled is accessed over HTTP rather than HTTPS, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Require authentication for API access

Description

Select this option when you want to require an API to authenticate before it can access a product. This option is selected by default.

When an API with this property enabled is accessed with invalid credentials, the security filter returns the HTTP status code 401 (unauthorized).



Note: For the Unica Platform `login` API, this configuration property is disabled, as this API is the first to be called for API authentication.

Default value

(Disabled)

Authentication mode

Description

Select this option when you want to authenticate API with Basic authentication or Bearer token authentication. When basic authentication or bearer token is selected, the corresponding user id and password has to be maintained in user's data source. For Manager authentication mode, it behaves in same way using parameter `api_auth_mode = Manager` in request header. This drop down selection is valid only if 'Require authentication for API access' is selected.

Default value

Manager

Data source credential holder

Description

Specify user name, which contains data source with required authentication credentials. Data source contains user id and password in case Basic authentication is selected in the Authentication mode drop down list. Data source contains bearer token in case Bearer token is selected in the Authentication mode drop down list.

Default value

asm_admin

Data source

Description

Specify data source name which is created under user specified in 'Data source credential holder'.

Default value

API_SECRET_DS

Related information

[Security framework for Unica APIs \(on page 186\)](#)

Unica Platform | Security | API management | [Product] | Unica Platform | User

(Affinium|suite|security|apiSecurity|manager|managerUser) Use the templates in this category to configure authentication for Unica APIs. You can block access, require HTTPS, or require authentication for APIs.

API URI

Description

For each product, the first part of the URI is resolved by the security framework, as follows:

```
http[s]://host:port/context_root/api/product
```

Therefore, in this field you should enter only the resource name or names of the API you want to configure. You can obtain the string you need to enter from the product's API documentation.

The value used for this property must start with a / (forward slash); otherwise the configuration is ignored by the security framework.

This property supports an exact URL match as well a pattern match for the configured APIs.

- For an exact match, the URI may end with a forward slash (/) or the resource name.
- For a pattern match, the URI must end with an asterisk (*).

If you set the value of this property to `/*` the settings you use for the other properties in the category apply to all APIs for the product.



Note: For the Unica Platform `login` API, this configuration property is read-only.

Default value

`/user/partitions/*`

Block API access

Description

Select this option when you want to prevent an API from accessing a product. This option is not selected by default.

When an API is blocked, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Secure API access over HTTPS

Description

Select this option when you want to allow the API to access a product only over HTTPS. This option is selected by default.

When an API with this property enabled is accessed over HTTP rather than HTTPS, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Enabled)

Require authentication for API access

Description

Select this option when you want to require an API to authenticate before it can access a product. This option is selected by default.

When an API with this property enabled is accessed with invalid credentials, the security filter returns the HTTP status code 401 (unauthorized).



Note: For the Unica Platform `login` API, this configuration property is disabled, as this API is the first to be called for API authentication.

Default value

(Enabled)

Authentication mode

Description

Select this option when you want to authenticate API with Basic authentication or Bearer token authentication. When basic authentication or bearer token is selected, the corresponding user id and password has to be maintained in user's data source. For Manager authentication mode, it behaves in same way using parameter `api_auth_mode = Manager` in request header. This drop down selection is valid only if 'Require authentication for API access' is selected.

Default value

Manager

Data source credential holder

Description

Specify user name, which contains data source with required authentication credentials. Data source contains user id and password in case Basic authentication is selected in the Authentication mode drop down list. Data source contains bearer token in case Bearer token is selected in the Authentication mode drop down list.

Default value

asm_admin

Data source**Description**

Specify data source name which is created under user specified in 'Data source credential holder'.

Default value

API_SECRET_DS

 Related information

[Security framework for Unica APIs \(on page 186\)](#)

Unica Platform | Security | API management | [Product] | Unica Platform | Policy

(Affinium|suite|security|apiSecurity|manager|managerPolicy)Use the templates in this category to configure authentication for Unica APIs. You can block access, require HTTPS, or require authentication for APIs.

API URI**Description**

For each product, the first part of the URI is resolved by the security framework, as follows:

```
http[s]://host:port/context root/api/product
```

Therefore, in this field you should enter only the resource name or names of the API you want to configure. You can obtain the string you need to enter from the product's API documentation.

The value used for this property must start with a / (forward slash); otherwise the configuration is ignored by the security framework.

This property supports an exact URL match as well a pattern match for the configured APIs.

- For an exact match, the URI may end with a forward slash (/) or the resource name.
- For a pattern match, the URI must end with an asterisk (*).

If you set the value of this property to `/*` the settings you use for the other properties in the category apply to all APIs for the product.



Note: For the Unica Platform `login` API, this configuration property is read-only.

Default value

/policy/partitions/*

Block API access

Description

Select this option when you want to prevent an API from accessing a product. This option is not selected by default.

When an API is blocked, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Secure API access over HTTPS

Description

Select this option when you want to allow the API to access a product only over HTTPS. This option is selected by default.

When an API with this property enabled is accessed over HTTP rather than HTTPS, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Require authentication for API access

Description

Select this option when you want to require an API to authenticate before it can access a product. This option is selected by default.

When an API with this property enabled is accessed with invalid credentials, the security filter returns the HTTP status code 401 (unauthorized).



Note: For the Unica Platform `login` API, this configuration property is disabled, as this API is the first to be called for API authentication.

Default value

(Enabled)

Authentication mode

Description

Select this option when you want to authenticate API with Basic authentication or Bearer token authentication. When basic authentication or bearer token is selected, the corresponding user id and password has to be maintained in user's data source. For Manager authentication mode, it behaves in

same way using parameter `api_auth_mode = Manager` in request header. This drop down selection is valid only if 'Require authentication for API access' is selected.

Default value

Manager

Data source credential holder

Description

Specify user name, which contains data source with required authentication credentials. Data source contains user id and password in case Basic authentication is selected in the Authentication mode drop down list. Data source contains bearer token in case Bearer token is selected in the Authentication mode drop down list.

Default value

asm_admin

Data source

Description

Specify data source name which is created under user specified in 'Data source credential holder'.

Default value

API_SECRET_DS

Related information

[Security framework for Unica APIs \(on page 186\)](#)

Unica Platform | Security | API management | [Product] | Unica Platform | Configuration

(Affinium|suite|security|apiSecurity|manager|managerConfiguration) Use the templates in this category to configure authentication for Unica APIs. You can block access, require HTTPS, or require authentication for APIs.

API URI

Description

For each product, the first part of the URI is resolved by the security framework, as follows:

```
http[s]://host:port/context_root/api/product
```

Therefore, in this field you should enter only the resource name or names of the API you want to configure. You can obtain the string you need to enter from the product's API documentation.

The value used for this property must start with a / (forward slash); otherwise the configuration is ignored by the security framework.

This property supports an exact URL match as well a pattern match for the configured APIs.

- For an exact match, the URI may end with a forward slash (/) or the resource name.
- For a pattern match, the URI must end with an asterisk (*).

If you set the value of this property to `/*` the settings you use for the other properties in the category apply to all APIs for the product.



Note: For the Unica Platform `login` API, this configuration property is read-only.

Default value

`/datasource/config`

Block API access

Description

Select this option when you want to prevent an API from accessing a product. This option is not selected by default.

When an API is blocked, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Secure API access over HTTPS

Description

Select this option when you want to allow the API to access a product only over HTTPS. This option is selected by default.

When an API with this property enabled is accessed over HTTP rather than HTTPS, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Require authentication for API access

Description

Select this option when you want to require an API to authenticate before it can access a product. This option is selected by default.

When an API with this property enabled is accessed with invalid credentials, the security filter returns the HTTP status code 401 (unauthorized).



Note: For the Unica Platform `login` API, this configuration property is disabled, as this API is the first to be called for API authentication.

Default value

(Enabled)

Authentication mode**Description**

Select this option when you want to authenticate API with Basic authentication or Bearer token authentication. When basic authentication or bearer token is selected, the corresponding user id and password has to be maintained in user's data source. For Manager authentication mode, it behaves in same way using parameter `api_auth_mode = Manager` in request header. This drop down selection is valid only if 'Require authentication for API access' is selected.

Default value

Manager

Data source credential holder**Description**

Specify user name, which contains data source with required authentication credentials. Data source contains user id and password in case Basic authentication is selected in the Authentication mode drop down list. Data source contains bearer token in case Bearer token is selected in the Authentication mode drop down list.

Default value

asm_admin

Data source**Description**

Specify data source name which is created under user specified in 'Data source credential holder'.

Default value

API_SECRET_DS

Related information

[Security framework for Unica APIs \(on page 186\)](#)

Unica Platform | Security | API management | [Product] | Unica Platform | Datasource

(Affinium|suite|security|apiSecurity|manager|managerDatasource)Use the templates in this category to configure authentication for Unica APIs. You can block access, require HTTPS, or require authentication for APIs.

API URI

Description

For each product, the first part of the URI is resolved by the security framework, as follows:

```
http[s]://host:port/context_root/api/product
```

Therefore, in this field you should enter only the resource name or names of the API you want to configure. You can obtain the string you need to enter from the product's API documentation.

The value used for this property must start with a / (forward slash); otherwise the configuration is ignored by the security framework.

This property supports an exact URL match as well a pattern match for the configured APIs.

- For an exact match, the URI may end with a forward slash (/) or the resource name.
- For a pattern match, the URI must end with an asterisk (*).

If you set the value of this property to `/*` the settings you use for the other properties in the category apply to all APIs for the product.



Note: For the Unica Platform `login` API, this configuration property is read-only.

Default value

/datasource

Block API access

Description

Select this option when you want to prevent an API from accessing a product. This option is not selected by default.

When an API is blocked, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Secure API access over HTTPS

Description

Select this option when you want to allow the API to access a product only over HTTPS. This option is selected by default.

When an API with this property enabled is accessed over HTTP rather than HTTPS, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Enabled)

Require authentication for API access**Description**

Select this option when you want to require an API to authenticate before it can access a product. This option is selected by default.

When an API with this property enabled is accessed with invalid credentials, the security filter returns the HTTP status code 401 (unauthorized).



Note: For the Unica Platform `login` API, this configuration property is disabled, as this API is the first to be called for API authentication.

Default value

(Enabled)

Authentication mode**Description**

Select this option when you want to authenticate API with Basic authentication or Bearer token authentication. When basic authentication or bearer token is selected, the corresponding user id and password has to be maintained in user's data source. For Manager authentication mode, it behaves in same way using parameter `api_auth_mode = Manager` in request header. This drop down selection is valid only if 'Require authentication for API access' is selected.

Default value

Manager

Data source credential holder**Description**

Specify user name, which contains data source with required authentication credentials. Data source contains user id and password in case Basic authentication is selected in the Authentication mode drop down list. Data source contains bearer token in case Bearer token is selected in the Authentication mode drop down list.

Default value

asm_admin

Data source**Description**

Specify data source name which is created under user specified in 'Data source credential holder'.

Default value

API_SECRET_DS

 Related information

[Security framework for Unica APIs \(on page 186\)](#)

Unica Platform | Security | API management | [Product] | Unica Platform | Login

(Affinium|suite|security|apiSecurity|manager|managerLogin)Use the templates in this category to configure authentication for Unica APIs. You can block access, require HTTPS, or require authentication for APIs.

API URI**Description**

For each product, the first part of the URI is resolved by the security framework, as follows:

```
http[s]://host:port/context root/api/product
```

Therefore, in this field you should enter only the resource name or names of the API you want to configure. You can obtain the string you need to enter from the product's API documentation.

The value used for this property must start with a / (forward slash); otherwise the configuration is ignored by the security framework.

This property supports an exact URL match as well a pattern match for the configured APIs.

- For an exact match, the URI may end with a forward slash (/) or the resource name.
- For a pattern match, the URI must end with an asterisk (*).

If you set the value of this property to `/*` the settings you use for the other properties in the category apply to all APIs for the product.



Note: For the Unica Platform `login` API, this configuration property is read-only.

Default value

/authentication/v1/login

Block API access

Description

Select this option when you want to prevent an API from accessing a product. This option is not selected by default.

When an API is blocked, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Secure API access over HTTPS

Description

Select this option when you want to allow the API to access a product only over HTTPS. This option is selected by default.

When an API with this property enabled is accessed over HTTP rather than HTTPS, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Require authentication for API access

Description

Select this option when you want to require an API to authenticate before it can access a product. This option is selected by default.

When an API with this property enabled is accessed with invalid credentials, the security filter returns the HTTP status code 401 (unauthorized).



Note: For the Unica Platform `login` API, this configuration property is disabled, as this API is the first to be called for API authentication.

Default value

(Disabled)

Authentication mode

Description

Select this option when you want to authenticate API with Basic authentication or Bearer token authentication. When basic authentication or bearer token is selected, the corresponding user id and password has to be maintained in user's data source. For Manager authentication mode, it behaves in

same way using parameter `api_auth_mode = Manager` in request header. This drop down selection is valid only if 'Require authentication for API access' is selected.

Default value

Manager

Data source credential holder**Description**

Specify user name, which contains data source with required authentication credentials. Data source contains user id and password in case Basic authentication is selected in the Authentication mode drop down list. Data source contains bearer token in case Bearer token is selected in the Authentication mode drop down list.

Default value

asm_admin

Data source**Description**

Specify data source name which is created under user specified in 'Data source credential holder'.

Default value

API_SECRET_DS

Related information

[Security framework for Unica APIs \(on page 186\)](#)

Unica Platform | Security | API management | [Product] | Unica Marketing Campaign | Interact Collection

(Affinium|suite|security|apiSecurity|campaign|Interact Collection)Use the templates in this category to configure authentication for Unica APIs. You can block access, require HTTPS, or require authentication for APIs.

API URI**Description**

For each product, the first part of the URI is resolved by the security framework, as follows:

```
http[s]://host:port/context_root/api/product
```

Therefore, in this field you should enter only the resource name or names of the API you want to configure. You can obtain the string you need to enter from the product's API documentation.

The value used for this property must start with a / (forward slash); otherwise the configuration is ignored by the security framework.

This property supports an exact URL match as well a pattern match for the configured APIs.

- For an exact match, the URI may end with a forward slash (/) or the resource name.
- For a pattern match, the URI must end with an asterisk (*).

If you set the value of this property to `/*` the settings you use for the other properties in the category apply to all APIs for the product.



Note: For the Unica Platform `login` API, this configuration property is read-only.

Default value

`/rest/v1/interactCollection/*`

Block API access

Description

Select this option when you want to prevent an API from accessing a product. This option is not selected by default.

When an API is blocked, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Secure API access over HTTPS

Description

Select this option when you want to allow the API to access a product only over HTTPS. This option is selected by default.

When an API with this property enabled is accessed over HTTP rather than HTTPS, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Require authentication for API access

Description

Select this option when you want to require an API to authenticate before it can access a product. This option is selected by default.

When an API with this property enabled is accessed with invalid credentials, the security filter returns the HTTP status code 401 (unauthorized).



Note: For the Unica Platform `login` API, this configuration property is disabled, as this API is the first to be called for API authentication.

Default value

(Disabled)

Authentication mode**Description**

Select this option when you want to authenticate API with Basic authentication or Bearer token authentication. When basic authentication or bearer token is selected, the corresponding user id and password has to be maintained in user's data source. For Manager authentication mode, it behaves in same way using parameter `api_auth_mode = Manager` in request header. This drop down selection is valid only if 'Require authentication for API access' is selected.

Default value

Manager

Data source credential holder**Description**

Specify user name, which contains data source with required authentication credentials. Data source contains user id and password in case Basic authentication is selected in the Authentication mode drop down list. Data source contains bearer token in case Bearer token is selected in the Authentication mode drop down list.

Default value

asm_admin

Data source**Description**

Specify data source name which is created under user specified in 'Data source credential holder'.

Default value

API_SECRET_DS

Related information

[Security framework for Unica APIs \(on page 186\)](#)

Unica Platform | Security | API management | [Product] | Unica Marketing Campaign | Triggered Messages

(Affinium|suite|security|apiSecurity|campaign|Interact Collection)Use the templates in this category to configure authentication for Unica APIs. You can block access, require HTTPS, or require authentication for APIs.

API URI

Description

For each product, the first part of the URI is resolved by the security framework, as follows:

```
http[s]://host:port/context_root/api/product
```

Therefore, in this field you should enter only the resource name or names of the API you want to configure. You can obtain the string you need to enter from the product's API documentation.

The value used for this property must start with a / (forward slash); otherwise the configuration is ignored by the security framework.

This property supports an exact URL match as well a pattern match for the configured APIs.

- For an exact match, the URI may end with a forward slash (/) or the resource name.
- For a pattern match, the URI must end with an asterisk (*).

If you set the value of this property to `/*` the settings you use for the other properties in the category apply to all APIs for the product.



Note: For the Unica Platform `login` API, this configuration property is read-only.

Default value

```
/rest/v1/triggeredMessages/*
```

Block API access

Description

Select this option when you want to prevent an API from accessing a product. This option is not selected by default.

When an API is blocked, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Secure API access over HTTPS

Description

Select this option when you want to allow the API to access a product only over HTTPS. This option is selected by default.

When an API with this property enabled is accessed over HTTP rather than HTTPS, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Require authentication for API access**Description**

Select this option when you want to require an API to authenticate before it can access a product. This option is selected by default.

When an API with this property enabled is accessed with invalid credentials, the security filter returns the HTTP status code 401 (unauthorized).



Note: For the Unica Platform `login` API, this configuration property is disabled, as this API is the first to be called for API authentication.

Default value

(Disabled)

Authentication mode**Description**

Select this option when you want to authenticate API with Basic authentication or Bearer token authentication. When basic authentication or bearer token is selected, the corresponding user id and password has to be maintained in user's data source. For Manager authentication mode, it behaves in same way using parameter `api_auth_mode = Manager` in request header. This drop down selection is valid only if 'Require authentication for API access' is selected.

Default value

Manager

Data source credential holder**Description**

Specify user name, which contains data source with required authentication credentials. Data source contains user id and password in case Basic authentication is selected in the Authentication mode drop down list. Data source contains bearer token in case Bearer token is selected in the Authentication mode drop down list.

Default value

asm_admin

Data source**Description**

Specify data source name which is created under user specified in 'Data source credential holder'.

Default value

API_SECRET_DS

 Related information

[Security framework for Unica APIs \(on page 186\)](#)

Unica Platform | Security | API management | [Product] | Unica Marketing Campaign | Campaign REST API Filter

(Affinium|suite|security|apiSecurity|campaign|Campaign REST API Filter)Use the templates in this category to configure authentication for Unica APIs. You can block access, require HTTPS, or require authentication for APIs.

API URI**Description**

For each product, the first part of the URI is resolved by the security framework, as follows:

```
http[s]://host:port/context_root/api/product
```

Therefore, in this field you should enter only the resource name or names of the API you want to configure. You can obtain the string you need to enter from the product's API documentation.

The value used for this property must start with a / (forward slash); otherwise the configuration is ignored by the security framework.

This property supports an exact URL match as well a pattern match for the configured APIs.

- For an exact match, the URI may end with a forward slash (/) or the resource name.
- For a pattern match, the URI must end with an asterisk (*).

If you set the value of this property to `/*` the settings you use for the other properties in the category apply to all APIs for the product.



Note: For the Unica Platform `login` API, this configuration property is read-only.

Default value

/rest/v1/*

Block API access

Description

Select this option when you want to prevent an API from accessing a product. This option is not selected by default.

When an API is blocked, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Secure API access over HTTPS

Description

Select this option when you want to allow the API to access a product only over HTTPS. This option is selected by default.

When an API with this property enabled is accessed over HTTP rather than HTTPS, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Require authentication for API access

Description

Select this option when you want to require an API to authenticate before it can access a product. This option is selected by default.

When an API with this property enabled is accessed with invalid credentials, the security filter returns the HTTP status code 401 (unauthorized).



Note: For the Unica Platform `login` API, this configuration property is disabled, as this API is the first to be called for API authentication.

Default value

(Enabled)

Authentication mode

Description

Select this option when you want to authenticate API with Basic authentication or Bearer token authentication. When basic authentication or bearer token is selected, the corresponding user id and password has to be maintained in user's data source. For Manager authentication mode, it behaves in

same way using parameter `api_auth_mode = Manager` in request header. This drop down selection is valid only if 'Require authentication for API access' is selected.

Default value

Manager

Data source credential holder

Description

Specify user name, which contains data source with required authentication credentials. Data source contains user id and password in case Basic authentication is selected in the Authentication mode drop down list. Data source contains bearer token in case Bearer token is selected in the Authentication mode drop down list.

Default value

asm_admin

Data source

Description

Specify data source name which is created under user specified in 'Data source credential holder'.

Default value

API_SECRET_DS

Related information

[Security framework for Unica APIs \(on page 186\)](#)

Unica Platform | Security | API management | [Product] | Unica Marketing Campaign | Engage REST API Filter

(Affinium|suite|security|apiSecurity|campaign|Engage REST API Filter)Use the templates in this category to configure authentication for Unica APIs. You can block access, require HTTPS, or require authentication for APIs.

API URI

Description

For each product, the first part of the URI is resolved by the security framework, as follows:

```
http[s]://host:port/context_root/api/product
```

Therefore, in this field you should enter only the resource name or names of the API you want to configure. You can obtain the string you need to enter from the product's API documentation.

The value used for this property must start with a / (forward slash); otherwise the configuration is ignored by the security framework.

This property supports an exact URL match as well a pattern match for the configured APIs.

- For an exact match, the URI may end with a forward slash (/) or the resource name.
- For a pattern match, the URI must end with an asterisk (*).

If you set the value of this property to `/*` the settings you use for the other properties in the category apply to all APIs for the product.



Note: For the Unica Platform `login` API, this configuration property is read-only.

Default value

`/rest/engage/*`

Block API access

Description

Select this option when you want to prevent an API from accessing a product. This option is not selected by default.

When an API is blocked, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Secure API access over HTTPS

Description

Select this option when you want to allow the API to access a product only over HTTPS. This option is selected by default.

When an API with this property enabled is accessed over HTTP rather than HTTPS, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Require authentication for API access

Description

Select this option when you want to require an API to authenticate before it can access a product. This option is selected by default.

When an API with this property enabled is accessed with invalid credentials, the security filter returns the HTTP status code 401 (unauthorized).



Note: For the Unica Platform `login` API, this configuration property is disabled, as this API is the first to be called for API authentication.

Default value

(Disabled)

Authentication mode**Description**

Select this option when you want to authenticate API with Basic authentication or Bearer token authentication. When basic authentication or bearer token is selected, the corresponding user id and password has to be maintained in user's data source. For Manager authentication mode, it behaves in same way using parameter `api_auth_mode = Manager` in request header. This drop down selection is valid only if 'Require authentication for API access' is selected.

Default value

Manager

Data source credential holder**Description**

Specify user name, which contains data source with required authentication credentials. Data source contains user id and password in case Basic authentication is selected in the Authentication mode drop down list. Data source contains bearer token in case Bearer token is selected in the Authentication mode drop down list.

Default value

asm_admin

Data source**Description**

Specify data source name which is created under user specified in 'Data source credential holder'.

Default value

API_SECRET_DS

Related information

[Security framework for Unica APIs \(on page 186\)](#)

Unica Platform | Security | API management | [Product] | Unica Marketing Campaign | Campaign REST API V2 Filter

(Affinium|suite|security|apiSecurity|campaign|Campaign REST API V2 Filter)Use the templates in this category to configure authentication for Unica APIs. You can block access, require HTTPS, or require authentication for APIs.

API URI

Description

For each product, the first part of the URI is resolved by the security framework, as follows:

```
http[s]://host:port/context_root/api/product
```

Therefore, in this field you should enter only the resource name or names of the API you want to configure. You can obtain the string you need to enter from the product's API documentation.

The value used for this property must start with a / (forward slash); otherwise the configuration is ignored by the security framework.

This property supports an exact URL match as well a pattern match for the configured APIs.

- For an exact match, the URI may end with a forward slash (/) or the resource name.
- For a pattern match, the URI must end with an asterisk (*).

If you set the value of this property to `/*` the settings you use for the other properties in the category apply to all APIs for the product.



Note: For the Unica Platform `login` API, this configuration property is read-only.

Default value

`/rest/v2/*`

Block API access

Description

Select this option when you want to prevent an API from accessing a product. This option is not selected by default.

When an API is blocked, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Secure API access over HTTPS

Description

Select this option when you want to allow the API to access a product only over HTTPS. This option is selected by default.

When an API with this property enabled is accessed over HTTP rather than HTTPS, the security filter returns the HTTP status code 403 (forbidden).

Default value

(Disabled)

Require authentication for API access

Description

Select this option when you want to require an API to authenticate before it can access a product. This option is selected by default.

When an API with this property enabled is accessed with invalid credentials, the security filter returns the HTTP status code 401 (unauthorized).



Note: For the Unica Platform `login` API, this configuration property is disabled, as this API is the first to be called for API authentication.

Default value

(Enabled)

Authentication mode

Description

Select this option when you want to authenticate API with Basic authentication or Bearer token authentication. When basic authentication or bearer token is selected, the corresponding user id and password has to be maintained in user's data source. For Manager authentication mode, it behaves in same way using parameter `api_auth_mode = Manager` in request header. This drop down selection is valid only if 'Require authentication for API access' is selected.

Default value

Manager

Data source credential holder

Description

Specify user name, which contains data source with required authentication credentials. Data source contains user id and password in case Basic authentication is selected in the Authentication mode drop down list. Data source contains bearer token in case Bearer token is selected in the Authentication mode drop down list.

Default value

asm_admin

Data source

Description

Specify data source name which is created under user specified in 'Data source credential holder'.

Default value

API_SECRET_DS

Related information

[Security framework for Unica APIs \(on page 186\)](#)

Platform | Security | JWT authentication

JWT authentication is used for Journey Designer+Unica Campaign. JWT authentication allows single sign-on between applications.

Enable JWT authentication

Description

When the check box for this property is selected, JWT authentication is enabled.

This property applies only in environments where Journey Designer is integrated with Unica Campaign.

Default value

disabled

JWT service URL

Description

The URL of the JWT service. This value differs, depending on whether you have applied Unica Platform FixPack 10.0.0.1. Refer to the following examples.

- If you have **not** applied FixPack 10.0.0.1:

`http://IP_ADDRESS/jwt/api/v1/tokens`

- If you have applied FixPack 10.0.0.1:

`http://IP_ADDRESS/api/v1/keys`

This property applies only in environments where Journey Designer is integrated with Unica Campaign.

JWT shared secret

Description

The shared secret key that is sent from Unica Platform to the JWT service for authentication. This key is shared between Unica Platform and Journey Designer. The JWT issuer is mapped to the JWT shared secret within the JWT service.

This property applies only in environments where Journey Designer is integrated with Unica Campaign, and where Unica Platform is version 10.0.0.0 (that is where Unica Platform FixPack 10.0.0.1 is **not** applied).

JWT issuer

Description

The issuer name and version that is sent from Unica Platform to the JWT service for authentication.

This property applies only in environments where Journey Designer is integrated with Unica Campaign.

Platform | Notifications

Properties in this category control system behavior for notifications that Unica 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

Platform | Audit Events

The property on this page determines whether audit events are tracked.

Is event auditing enabled?

Description

Specifies whether the audit events are tracked.

Default value

False

Valid values

True | False

Platform | Audit Events | Audit events configuration

The events you select on this page are available in the security audit reports.

Record login and logout events for all accounts

Description

Specifies whether to track the user name and the date and time for log in and log out events for all user accounts.

Record when user sessions time out for all accounts

Description

Specifies whether to track the account user name and the date and time of sessions that are automatically timed out.

Record login and logout events for members of the HighSeverityAccounts group

Description

Specifies whether to track the user name and the date and time for log in and log out events for accounts that are members of the **highSeverityAccounts** group in Unica Platform. To enable this feature, you must set a severity level for this configuration property and add users to the highSeverityAccounts group.

Record LDAP group membership changes

Description

Specifies whether to record the addition or deletion of accounts, along with the user names and dates and times of these actions, for user accounts synchronized from an LDAP server. This property applies only when Unica Platform is integrated with a supported LDAP server, such as IBM Security Directory server or Windows™ Active Directory.

Record when accounts are enabled and disabled

Description

Specifies whether to record the account user name and the date and time when user accounts are enabled or disabled.

Record when account passwords change

Description

Specifies whether to record the account user name and the date and time when user passwords change.

Record when account passwords are locked

Description

Specifies whether to record the account user name and the date and time when a password is locked out due to too many incorrect login attempts.

Record when groups are created or deleted in Platform

Description

Specifies whether to record when groups are added or deleted.

Record Platform group membership changes

Description

Specifies whether to record when user accounts are added to or removed from a group.

Record Platform group permission changes

Description

Specifies whether to record changes to group permissions.

Record role creation or deletion

Description

Specifies whether to record when roles are added or deleted. Only roles that are shown on the **Settings > User roles and permissions** page are tracked.

Record role membership changes

Description

Specifies whether to record changes in role membership. Only roles that are shown on the **Settings > User roles and permissions** page are tracked.

Record role permission changes

Description

Specifies whether to record changes in role permissions. Only roles that are shown on the **Settings > User roles and permissions** page are tracked.

Record changes to properties on the configuration page

Description

Specifies whether to record changes in configuration properties on the **Settings > Configuration** page. Changes made by users on the Configuration page, or by users running the `configTool` are tracked. Configuration changes made by the installers during installation or upgrade are not tracked.

Enable audit backup

Description

Specifies whether to save audit data to the `USM_AUDIT_BACKUP` table.



Important: Because this is a bootstrap property that is read when the Unica Platform web application starts, you must stop and restart the Unica Platform web application when you change this property value.

Default value

False

Valid values

True | False

Archive data after the number of days specified here

Description

Specifies the interval, in days, between audit backups. The archived data is stored in the `USM_AUDIT_BACKUP` table and can be included in the Audit Events report when you set a custom date range that includes data from the archive.




Important: Because this is a bootstrap property that is read when the Unica Platform web application starts, you must stop and restart the Unica Platform web application when you change this property value.

Keep Audit records in primary for number days specified here

Description


Specifies how many days of data to keep in the `USM_AUDIT` table for the Audit Events report. When the default settings for the Audit Events report are in effect, only the data in the `USM_AUDIT` table is shown in the report.

 **Important:** Because this is a bootstrap property that is read when the Unica Platform web application starts, you must stop and restart the Unica Platform web application when you change this property value.

Archive start time

Description

Specifies the time of day when the system moves audit data into an archive. Use the 24 hour format for this value.

 **Important:** Because this is a bootstrap property that is read when the Unica Platform web application starts, you must stop and restart the Unica Platform web application when you change this property value.

Name of group to receive audit backup notifications

Description

Specifies the Unica group whose members should receive notification of archive backups. You can specify only one group for this property. Users in this group can manage their subscription to this notification by going to their **Settings > Users** page and clicking **Notification subscriptions**.

Platform | Audit Events | Audit events severity configuration

The severity level you specify for each event on this page appears in the Audit Events report. You can use the severity level to sort and filter the report data. The events are identical to those in the **Platform | Audit Events | Audit events configuration** category.

Digital Analytics configuration properties

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

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

Digital Analytics®

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

Enable Coremetrics® Analytics

Description

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

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

See the Unica Platform Administrator's Guide for details about this integration.

Default value

`false`

Digital Analytics® | Integration | partitions | partition[n]

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

Platform user for Coremetrics® account

Description

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

This is part of the configuration for enabling single sign-on between Digital Analytics and Unica. See the Unica 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 Digital Analytics shared secret.

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

Default value

`CoremetricsDS`

Report configuration properties

The report configuration properties for Unica are available at **Settings > Configuration > Reports**.

To generate reports, the Unica suite integrates with Cognos®, a business intelligence application. You use the **Integrations > Cognos** properties to identify your Cognos® system. Then, for Unica Campaign, Unica Deliver, and

Unica Interact, you must configure additional properties to set up and customize the reporting schemas. For more details on the configuration properties, see the Cognos Reports Installation and Configuration Guide.

Unica Plan configuration properties

The Unica Plan configuration properties are available on the **Settings > Configuration** page. For more details on the configuration properties, see the Plan Administrator Guide.

Unica Campaign configuration properties

The configuration properties of Unica Campaign are located at **SettingsConfiguration**. For more details on the configuration properties, see the Campaign Administrator Guide.

Unica Deliver configuration properties

The Unica Deliver configuration properties are available on the Configuration page. For more details on the configuration properties, see the Deliver Startup and Administrator Guide.

Unica Interact configuration properties

The Unica Interact configuration properties are available on the Configuration page. For more details on the configuration properties, see the Interact Administrator Guide.

Unica Journey configuration properties

The Unica Journey configuration properties are available on the **Settings > Configuration** page. For more details on the configuration properties, see the Journey Administrator Guide.

Unica Content Integration configuration properties

The Unica Content Integration configuration properties are available on the **Settings > Configuration** page. For more details on the configuration properties, see the Content Integration Administrator Guide.

Unica Collaborate configuration properties

The Unica Collaborate Integration configuration properties are available on the **Settings > Configuration** page. For more details on the configuration properties, see the Unica Collaborate Administrator Guide.

Additional configuration properties exist in XML files that are located under the Unica Collaborate installation directory.

IBM SPSS Modeler Advantage Enterprise Marketing Management Edition configuration properties

Properties in this category specify values that are used to configure Unica for single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

See the Unica Campaign and IBM SPSS Modeler Advantage Enterprise Marketing Management Edition Integration Guide for complete instructions on setting up single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

SPSS® | integration

Properties in this category are used to configure Unica Platform for single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

Platform user for IBM® SPSS® account

Description

Enter the login name for the IBM SPSS Modeler Advantage Enterprise Marketing Management Edition account that you created or identified for single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

Default value

asm_admin

Availability

This property is used only to configure Unica Platform for single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

Datasource for IBM® SPSS® account

Description

Set this property to the name of the data source you created for the system user when you configured single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition. If you used **SPSS_MA_ADMIN_DS** as the data source name, you can retain the default value of this property.

Default value

SPSS_MA_ADMIN_DS

Availability

This property is used only to configure Unica Platform for single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

Is this score only integration

Description

Not supported.

Default value

FALSE

Availability

This property is used only to configure Unica Platform for single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

SPSS® | integration | partitions | partition [n]

The property in this category is used to configure Unica Platform for single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

Enable IBM® SPSS®**Description**

Set this property to `TRUE` to enable single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

For each partition where you have users who should have single sign-on, you must use the **SPSS MA EMM Edition | Integration | partitions | partitionTemplate** to create the **enableSPSS** configuration property for that partition. The name of the category you create with the template must exactly match the name of the corresponding Campaign partition. The default partition1 already has the **Enable IBM SPSS** configuration property, so you do not have to use the template to create it.

Default value

`FALSE`

Availability

This property is used only to configure Unica Platform for single sign-on with IBM SPSS Modeler Advantage Enterprise Marketing Management Edition.

SPSS® | navigation

Properties in this category affect IBM SPSS Modeler Advantage Enterprise Marketing Management Edition integration with Unica 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 SPSS® decision management server. Configure this URL with server name or server IP address followed by the port on which IBM SPSS Modeler Advantage Enterprise Marketing Management Edition is hosted on the server.

Default value

One of the following formats:

- `http://<server name>:<port>/DM`
- `http://<server IP address>:<port>/DM`

Valid values

The URL for the SPSS® decision management server.

C&DS Server

Description

The name of the IBM SPSS Collaboration and Deployment Services server.

Default value

None

Valid values

Valid server name or server IP address on which IBM SPSS Collaboration and Deployment Services is installed and configured.

C&DS Port

Description

The port where the IBM SPSS Collaboration and Deployment Services server is located.

Default value

None

Valid values

Valid port number on which IBM SPSS Collaboration and Deployment Services is hosted.

Opportunity Detect and Unica Interact Advanced Patterns configuration properties

This section describes the Opportunity Detect and Unica Interact Advanced Patterns configuration properties on the Configuration page.

Opportunity Detect and Interact Advanced Patterns | Navigation

Properties in this category specify values that are used internally to navigate among Unica products.

welcomePageURI

Description

The Uniform Resource Identifier of the Opportunity Detect index page. This value is used internally by Unica applications. Changes to this value are not recommended.

Default value

/index.jsp

seedName

Description

Used internally by Unica applications. Changes to this value are not recommended.

Default value

Detect

type

Description

Used internally by Unica applications. Changes to this value are not recommended.

Default value

Detect

httpPort

Description

The port number that is used by the application server for connections to the Opportunity Detect application.

Default value

7001

httpsPort

Description

The port number that is used by the application server for secure connections to the Opportunity Detect application.

Default value

7001

serverURL

Description

The URL of the Opportunity Detect installation. Accepts either the HTTP or HTTPS protocol. If you are on a clustered environment and choose to use ports that are different from the default ports 80 or 443 for your deployment, do not use a port number in the value of this property.

If users access Opportunity Detect with the Chrome browser, use the fully qualified domain name (FQDN) in the URL. If the FQDN is not used, the Chrome browser cannot access the product URLs.



Important: If Unica 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.

Unica Platform uses this value to call the logout handler of each registered application if the user clicks the logout link in Unica.

serverURLInternal

Description

Used internally. Changes to this value are not recommended.

displayName

Description

Used internally. Changes to this value are not recommended.

Default value

Opportunity Detect

Opportunity Detect and Interact Advanced Patterns | System | Streams Remote Control Web Service

The property in this category specifies the URL for the InfoSphere Streams remote control web service. Opportunity Detect Design Time communicates with Opportunity Detect Run Time over this service.

ServerURL

Description

The person who installs the product sets this property value during installation. The default port number is 8080.

Default value

`http://[SRCSTHost]:[SRCSTPort]/axis2/services/RemoteControl`

Opportunity Detect and Interact Advanced Patterns | System | Real Time Connector

The property in this category specifies the URL for the web service used when Unica Interact is integrated with Unica Interact Advanced Patterns or when the Web Service connector is used for input data.

ServerURL

Description

The person who installs the product sets this property value during installation. The default port number is 8282.

Default value

```
http://[RealTimeConnectorHost]:[RealTimeConnectorPort]/servlets/StreamServlet
```

Opportunity Detect and Interact Advanced Patterns | System | Monitoring

Properties in this category specify values that affect the monitoring tool.

Poll Interval (In Seconds)

Description

The number of seconds that the monitoring service waits between two successive polls of the Streams server for the statistics. The default is 300 seconds, or 5 minutes.

Default value

```
300
```

Retaining Time (In Days)

Description

The number of days the monitoring service should keep the polled data in the database. The default is 10 days. Data that is older than the time specified here is purged.

Default value

```
10
```

Opportunity Detect and Interact Advanced Patterns | System | Processing Options

Properties in this category specify values that affect the monitoring tool.

Cache profile records

Description

Opportunity Detect can cache profile data, which provides optimal performance. To enable caching of profile data, set the value of this property to `True`.

If you have very large profile data sets, you might want to retain the default value of this property, which is `False`. This disables caching of profile data and eliminates the out of memory issues that caching large amounts of profile data can cause.

If you change this property value, you must restart your web application server, the Streams instance, and the StreamsRCS service, and redeploy all affected deployments.

Default value

`False`

Opportunity Detect and Interact Advanced Patterns | logging

The property in this category specifies the location of the Opportunity Detect log file.

log4jConfig

Description

The location of the configuration file that Opportunity Detect uses for logging. This value is set automatically during installation, but if you change this path, you must restart the web application server to apply the change.

Default value

`[absolute-path]/conf/detect_log4j.properties`

Unica Interact Advanced Patterns | System | Interact Design Service

The property in this category specifies the URL for the web service that allows Interact to automatically create and deploy advanced patterns when Unica Interact is integrated with Unica Interact Advanced Patterns.

ServerURL

Description

This web service is the integration point between Unica Interact and Unica Interact Advanced Patterns design time. The person who installs the product sets this property value during installation. The default port number is 8181.

Default value

`http://[InteractServiceHost]:[InteractServicePort]/axis2/services/InteractDesignService`

Here are the configuration properties laid down by the Installer.

Insights | navigation

The Unica suite integrates with Unica Insights to generate reports.

This page displays properties that specify URLs and other parameters that are used by the Unica Insights system.

Seed Name**Description**

Used internally by HCL Unica applications. Changes to this value are not recommended.

Default value

Insights

httpPort**Description**

This property specifies the port used by the Unica Insights web application server. If your installation of Unica Insights uses a port which 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 Unica Insights web application server for secure connections. If your installation of Unica Insights uses a secure port that is different from the default, you must edit the value of this property.

Default value

7001

serverURL**Description**

Specifies the URL of the Unica Insights web application. 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:7001/ Insights`

Default value

`http://[CHANGE ME]/hcl-birt`

Valid values

A well-formed URL

logoutURL**Description**

The `logoutURL` property is used internally to call the logout handler of the registered application if the user clicks the logout link. Do not change this value.

Default value

```
/j_spring_security_logout
```

Enabled

Description

Setting the value to `TRUE` ensures that Unica Insights will be used as reporting engine.



Note: If you are upgrading to V 12.0 and you have Campaign/Plan/Interact Reports pack and Unica Platform installed, then you can either see Cognos Reports or Unica Insights reports.

Default value

```
False
```

Valid values

```
FALSE | TRUE
```

Currently, Unica Insights reports are supported for Oracle, SQL Server, and DB2 databases.

Customization of stylesheets and images in the Unica user interface

You can customize the appearance of the user interface where most Unica 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 HCL logo and color scheme with your company's logo and color scheme.

Stylesheets

The 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.

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.

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 uses sprites, the name of the image includes `_sprites`. If you want to replace these images, you should use sprites with the same dimensions, as this requires the fewest modifications to the stylesheet. If you are not familiar with sprites, you can learn about them on the internet.

Preparing your corporate theme

Follow these guidelines to create your corporate theme for the Unica frameset.

1. When you installed Unica 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 Unica user interface.

1. Place the images you want to use (for example, your logo, buttons, and icons) in a directory accessible from the machine where Unica Platform is installed. Refer to the modified `corporatetheme.css` file, created as described in a "Preparing your corporate theme," to determine where to place your images.
2. If Unica Platform is deployed, undeploy it.
3. When you installed Unica Platform, you may have created an EAR file containing the `unica.war` file, or you may have installed the `unica.war` file. In either case, do the following.
 - Make a backup of your WAR or EAR file, saving the backup with a different name (for example, `original_unica.war`). This enables you to roll back your changes if necessary.
 - Extract your installed file as necessary to access the files and directories the `unica.war` contains.
4. Place the modified `corporatetheme.css` file, created as described in "Preparing your corporate theme," in the `css\theme` directory.

This overwrites the blank `corporatetheme.css` file that is already there.

5. Re-create the `unica.war` file, and, if necessary, the EAR file that contained it.
6. Deploy the WAR or EAR file.

7. Clear your browser cache and log in to Unica.

Your new theme should be applied.