Unica Platform V12.1.2 Upgrade Guide

Contents

Chapter 1. Upgrade overview

An upgrade of Unica Platform is complete when you upgrade, configure, and deploy Unica Platform. The Unica Platform Upgrade Guide provides detailed information about upgrading, configuring, and deploying Unica Platform.

Use the Upgrade roadmap section to obtain a broad understanding about using the Unica Platform Upgrade Guide.

Upgrade roadmap

Use the upgrade roadmap to quickly find the information that you need for installing Unica Platform.

You can use the following table to scan the tasks that must be completed for installing Unica Platform

Table 1. Unica Platform upgrade roadmap

This two-columned table describes the topics that are included in the Unica Platform Upgrade Guide in one column, and the explanation of the tasks in the second column.

Topic	Information
Upgrade overview *(on page 1)*	This topic provides the following information: • How the installers work *(on page 7)* • Modes of installation *(on page 8)* • Unica Platform documentation and help *(on page 11)*
Planning the Unica Platform upgrade *(on page 13)*	This topic provides the following information: • Prerequisites *(on page 13)* • Unica Platform installation worksheet *(on page 17)* • Upgrade prerequisites for all Unica products *(on page 20)* • Time zone support for scheduler *(on page 22)*

Table 1. Unica Platform upgrade roadmap

This two-columned table describes the topics that are included in the Unica Platform Upgrade Guide in one column, and the explanation of the tasks in the second column. (continued)

Topic	Information
	• Backing up the corporatetheme.css file and branding images *(on page 22)* • Unica Platform upgrade scenarios *(on page 23)*
Upgrading from version 10.0 and higher with automatic migration *(on page 28)*	Read more about how to upgrade Unica Platform from version 11.1 and higher with automatic migration. ⚠️ **Important:** If you are upgrading from a version prior to 11.1, you must first upgrade to version 11.1 and then upgrade to version 12.1. For information about upgrading to version 11.1, see the Unica Platform 11.1 Upgrade Guide.
• Upgrading from version 8.6.0 with manual migration *(on page 31)* • Upgrading from version 9.0.0 with manual migration *(on page 49)*	Read more about how to upgrade Unica Platform from version 8.6.0 and above with manual migration. Use the instructions in the chapter that corresponds with your current version.

Table 1. Unica Platform upgrade roadmap

This two-columned table describes the topics that are included in the Unica Platform Upgrade Guide in one column, and the explanation of the tasks in the second column.

(continued)

Topic	Information
• Upgrading from version 9.1.0 with manual migration *(on page 65)* • Upgrading from version 9.1.1 with manual migration *(on page 78)* • Upgrading from version 9.1.2 with manual migration *(on page 90)* • Upgrading from version 10.0.0 with manual migration	

Table 1. Unica Platform upgrade roadmap

This two-columned table describes the topics that are included in the Unica Platform Upgrade Guide in one column, and the explanation of the tasks in the second column.

(continued)

Topic	Information
(on page 101) • Upgrading from version 10.1.0 with manual migration *(on page 111)* • Upgrading from version 11.0.0 with manual migration *(on page 120)* • Upgrading from version 11.0.1 with manual migration *(on page 128)* • Upgrading from version 11.1.0	

Table 1. Unica Platform upgrade roadmap

This two-columned table describes the topics that are included in the Unica Platform Upgrade Guide in one column, and the explanation of the tasks in the second column.

(continued)

Topic	Information
with manu-al migration *(on page 136)* • Upgrading from ver-sion 12.0.0 with manu-al migration *(on page 144)* • Upgrading from ver-sion 12.1.0 with manu-al migration *(on page 152)* • Upgrading from ver-sion 12.1.1 with manu-al migration *(on page 159)*	

Table 1. Unica Platform upgrade roadmap

This two-columned table describes the topics that are included in the Unica Platform Upgrade Guide in one column, and the explanation of the tasks in the second column. (continued)

Topic	Information
Deploying Unica Platform *(on page 168)*	This topic provides the following information: • Guidelines for deploying Unica Platform on WebLogic *(on page 168)* • Guidelines for deploying Unica Platform on WebSphere *(on page 170)* • Guidelines for deploying Unica Platform on JBoss *(on page 172)* • Guidelines for deploying Unica Platform on Apache Tomcat *(on page 174)* • Verifying your Unica Platform installation *(on page 175)*
Configuring Unica Platform after deployment *(on page 177)*	This topic provides the following information: • Default password settings *(on page 178)*
Unica Platform utilities *(on page 179)*	This topic provides the following information: • Setting up Unica Platform utilities on additional machines *(on page 182)* • alertConfigTool *(on page 183)* • configTool *(on page 184)* • datafilteringScriptTool *(on page 190)* • encryptPasswords *(on page 192)* • encryptTomcatDBPasswords *(on page 194)* • partitionTool *(on page 195)* • populateDb *(on page 198)*

Table 1. Unica Platform upgrade roadmap

This two-columned table describes the topics that are included in the Unica Platform Upgrade Guide in one column, and the explanation of the tasks in the second column.

(continued)

Topic	Information
	• restoreAccess *(on page 199)* • scheduler_console_client *(on page 202)* • quartzjobtool *(on page 204)*
Unica Platform SQL scripts *(on page 205)*	This topic provides the following information: • ManagerSchema_DeleteAll.sql *(on page 205)* • ManagerSchema_PurgeDataFiltering.sql *(on page 206)* • SQL scripts for creating system tables *(on page 206)* • ManagerSchema_DropAll.sql *(on page 208)*.
Uninstalling Unica Platform *(on page 210)*	This topic provides information about how to uninstall Unica Platform.

How the installers work

Use the Unica installer with the Platform installers when you install Platform.

The Unica suite installer starts the individual product installers during the installation process.

Use the following guidelines to install Platform.

• Make sure that the Unica installer and the Platform installer are in the same directory on the server where you want to install Platform. When multiple versions of the Platform installer are present in the directory with the Unica installer, the Unica

installer shows the latest version of Platform on the **HCL Unica Products** screen in the installation wizard.

- If you are planning to install a patch immediately after you install Platform and Optimize, make sure that the patch installer is in the same directory as that of the Unica and the Platform installer.

Unica is installed in one of the following directories by default:

- `/opt/HCL/Unica` (for UNIX™)
- `C:\HCL\Unica` (for Windows™)

Unica products are installed by default in a subdirectory of the *HCL_home* directory. For example, Unica Platform is installed in the *HCL_Unica_home*`/Platform` directory.

Platform is installed in the `HCL_Unica_home/Campaign` directory by default.

Modes of installation

The Unica suite installer can run in one of the following modes: GUI mode, X Window System mode, console mode, or silent mode (also called the unattended mode). Select a mode that suits your requirements when you install Unica Platform.

For upgrades, you use the installer to perform many of the same tasks that you perform during the initial installation.

GUI X Window System mode

Use the GUI mode for Windows™ or the X Window System mode for UNIX™ to install Unica Platform by using the graphical user interface.

UNIX™ X Window System mode

Use the X Window System mode for UNIX™ to install Unica Platform by using the graphical user interface.

Console mode

Use the console mode to install Unica Platform by using the command line window.

> ✏️ **Note:** To display the Installer screens correctly in console mode, configure your
> terminal software to support UTF-8 character encoding. Other character encoding,
> such as ANSI, will not render the text correctly, and some information will not be
> readable.

Silent mode

Use the silent or unattended mode to install Unica Platform multiple times. The silent mode
uses response files for installation, and does not require user input during the installation
process.

> ✏️ **Note:** Silent mode is not supported for upgrade installations in clustered web
> application or clustered listener environments.

Sample response files

You must create response files to set up a silent installation of Unica Platform. You can use
sample response files to create your response files. The sample response files are included
with the installers in the `ResponseFiles` compressed archive.

The following table provides information about sample response files:

Table 2. Description of sample response files

Sample response file	Description
`installer.properties`	The sample response file for the Unica master installer.
`installer_product initials and product version number.properties`	The sample response file for the Unica Platform installer.
	For example, `installer_ucn.n.n.n.properties` is the response file of the Unica Campaign installer, where n.n.n.n is the version number.

Table 2. Description of sample response files

(continued)

Sample response file	Description
	For example, `installer_umpn.n.n.n.proper-ties` is the response file of the Platform installer, where n.n.n.n is the version number. For example, `installer_uln.n.n.n.properties` is the response file of the Leads installer, where n.n.n.n is the version number.
`installer_report pack initials, product initials, and version number.properties`	The sample response file for the reports pack installer. For example, `installer_urpcn.n.n.n.proper-ties` is the response file of the Unica Campaign reports pack installer, where n.n.n.n is the version number For example, `installer_urpl.properties` is the response file of the Leads reports pack installer.

Table 3. Description of sample response files

Sample response file	Description
`installer.properties`	The sample response file for the Unica master installer.
`installer_product in-tials and product version number.prop-erties`	The sample response file for the Unica Platform installer. For example, `installer_ucn.n.n.n.properties` is the response file of the Unica Campaign installer, where n.n.n.n is the version number. For example, `installer_umpn.n.n.n.properties` is the response file of the Unica Platform installer, where n.n.n.n is the version number.

Table 3. Description of sample response files

(continued)

Sample response file	Description
	For example, `installer_uln.n.n.n.properties` is the response file of the Leads installer, where n.n.n.n is the version number.

Unica Platform documentation and help

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

Table 4. Get up and running

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

Table 5. Configure and use Unica Platform

Task	Docu-mentation
• 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	*Unica Platform Administrator's Guide*

Chapter 2. Planning the Unica Platform upgrade

You can upgrade your current version of Unica Platform to bring it up-to-date with the latest features.

- Unica Platform supports the following upgrade paths:
    - 12.1.x → 12.1.2
    - 12.1.0.x → 12.1.2
- Customers on versions earlier than 8.6.x must:
    - perform a Fast Upgrade from existing version to version 8.6.0 (for more information, see *HCL Unica 8.6.0 Fast Upgrade Guide*).
    - perform a Fast Upgrade from version 8.6.0 to version 12.1.0 (for more information, see *HCL Unica 12.1.0 Fast Upgrade Guide*).
    - perform an in-place upgrade from version 12.1.0 to version 12.1.2.
- Customers on versions earlier than 11.1.x.x can:
    - perform a Fast Upgrade from existing version to version 12.1.0 (for more information, see *HCL Unica 12.1.0 Fast Upgrade Guide*).
    - perform an in-place upgrade from version 12.1.0 to version 12.1.2.
- Customers on versions 11.1.x.x/12.0.x.x can use one of the following options for upgrade:
    - **Option 1**
        - perform an in-place upgrade from existing version to version 12.1.0.
        - perform an in-place upgrade from version 12.1.0 to version 12.1.2.
    - **Option 2**
        - perform a Fast Upgrade from existing version to version 12.1.0 (for more information, see *HCL Unica 12.1.0 Fast Upgrade Guide*).
        - perform an in-place upgrade from version 12.1.0 to verison 12.1.2.

Prerequisites

Before you install or upgrade any Unica product, you must ensure that your computer complies with all of the prerequisite software and hardware.

System requirements

For information about system requirements, see the Recommended Software Environments and Minimum System Requirements guide.

Network domain requirements

The Unica products that are installed as a suite must be installed on the same network domain to comply with the browser restrictions that are designed to limit the security risks that can occur with cross-site scripting.

JVM requirements

Unica applications within a suite must be deployed on a dedicated Java™ virtual machine (JVM). Unica products customize the JVM that is used by the web application server. If you encounter errors that are related to the JVM, you must create an Oracle WebLogic or WebSphere®a WebSphere® domain that is dedicated to the Unica products.

Knowledge requirements

To install Unica products, you must have a thorough knowledge of the environment in which the products are installed. This knowledge includes knowledge about operating systems, databases, and web application servers.

Internet browser settings

Make sure that your internet browser complies with the following settings:

- The browser must not cache web pages.
- The browser must not block pop-up windows.

Access permissions

Verify that you have the following network permissions to complete the installation tasks:

- Administration access for all necessary databases

> **Note:** Administrator must have `CREATE`, `SELECT`, `INSERT`, `UPDATE`, `DELETE`, and `DROP` rights for both tables and views.

- Read and write access to the relevant directory and sub-directories for the operating system account that you use to run the web application server and Unica components
- Write permission for all files that you must edit
- Write permission for all directories where you must save a file, such as the installation directory and backup directory if you are upgrading
- Appropriate read, write, and execute permissions to run the installer
- Root access for the Linux™ operating system.

Verify that you have the administrative password for your web application server.

For UNIX™, all installer files for products must have full permissions, for example, rwxr-xr-x.

> ✎ **Note:** If you are performing a silent mode install, and you do not wish the system to prompt you for the SUDO Password, you could set a NOPASSWD for the SUDO user. If you do not set it, you are then prompted for the SUDO Password.For Windows installation - You can add the Unica Optimize installation user to the administrators group.

The following additional permissions are necessary for UNIX™:

- The user account that installs Campaign and Unica Platform must be a member of the same group as the Unica Campaign users. This user account must have a valid home directory, and have write permissions for that directory.
- All installer files for HCL Unica products must have full permissions, for example, rwxr-xr-x.

Points to consider before you install Unica Campaign

For Unica Campaign installation you are required to consider the following points.

JAVA_HOME environment variable

If a JAVA_HOME environment variable is defined on the computer where you install an Unica product, verify that the variable points to a supported version of JRE. For information

about system requirements, see the Unica Recommended Software Environments and Minimum System Requirements guide.

If the JAVA_HOME environment variable points to an incorrect JRE, you must clear the JAVA_HOME variable before you run the Unica installers.

You can clear the JAVA_HOME environment variable by using one of the following methods:

- Windows™: In a command window, enter set JAVA_HOME= (leave empty) and press **Enter**.
- UNIX™: In the terminal, enter export JAVA_HOME= (leave empty) and press **Enter**.

You can clear the JAVA_HOME environment variable by running the following command in the terminal:

export JAVA_HOME= (leave empty)

The Unica installer installs a JRE in the top-level directory for the Unica installation. Individual Unica application installers do not install a JRE. Instead, they point to the location of the JRE that is installed by the Unica installer. You can reset the environment variable after all installations are complete.

For more information about the supported JRE, see the Unica Recommended Software Environments and Minimum System Requirements guide.

Note: Ensure that you do not select the database type "Informix" during installation of versions 12.0.0.x and 12.10.0, since it is not functional.

Note: For versions 12.0.0 and higher, ensure that you do not select the database type Informix as it is not functional. From version 12.1.0.3 and higher, users can use OneDB database as system tables and user tables. See the *Unica V12.1.0.3 Installation Guide for OneDB* for more details.

Unica Platform installation worksheet

Use the Unica Platform installation worksheet to gather information about the Unica Platform database and about other Unica products that are required for the installation of Unica Platform.

Use the following table to gather information about the database that contains your Unica Platform system tables:

Table 6. Information about the database

This two-columned table provides a checklist of the various types of information that you must gather about the database in the first column and provides space to note the information in the second column.

Field	Notes®
Database type	
Database name	
Database account user name	
Database account password	
JNDI name	UnicaPlatformDS
ODBC name	

Refer the following information if user is planning to install Contact Central

Field	Notes®
Database type	
Database name	

Field	Notes®
Database account user name	
Database account password	
JNDI name	ContactCentral-DS
ODBC name	

Checklist for Unica Platform database

The installation wizards for each Unica product must be able to communicate with the Unica Platform system table database to register the product. Each time that you run the installer, you must enter the following database connection information for the Unica Platform system table database:

- Database type
- JDBC connection URL
- Database host name
- Database port
- Database name or schema ID
- User name and password for the database account

Checklist for Unica Platform deployment on the web application server

Obtain the following information before deploying Unica Platform:

- Protocol: HTTP or HTTPS if SSL is implemented in the web application server.
- Host: The name of the machine on which the Unica Platform will be deployed.
- Port: The port on which the web application server listens.
- Domain name: The company domain of each machine where HCL products are installed. For example, `mycompany.com`. All HCL products must be installed in the same company domain, and you must enter the domain name in all lower-case letters.

  If there is a mis-match in domain name entries, you may encounter problems when you attempt to use Unica Platform features or navigate among products. You can

change the domain name after the products are deployed by logging in and changing values of the relevant configuration properties in the product navigation categories on the **Settings > Configuration** page.

Checklist for enabling Unica Platform utilities

If you plan to use the Unica Platform utilities, obtain the following JDBC connection information before you start to install the Unica Platform.

- Path to the JRE. The default value is the path to the 1.8 version of the JRE that the installer places under your installation directory.

  You can accept this default or specify a different path. If you specify a different path, you must point to the 1.8 version of the Sun JRE.
- JDBC driver class. The installer automatically provides this, based on the database type you specifiy in the installer.
- JDBC connection URL. The installer provides the basic syntax, including the host name, database name, and port. You can customize the URL by adding additional parameters.
- JDBC driver classpath on your system.

Information about the web component

Obtain the following information for all Unica products that have a web component, which you deploy on a web application server:

- The name of the systems on which the web application servers are installed. You can have one or several web application servers, depending on the Unica environment that you are setting up.
- The port on which the application server listens. If you plan to implement SSL, obtain the SSL port.
- The network domain for your deployment system. For example, mycompany.com.

Upgrade prerequisites for all Unica products

Meet all requirements for permissions, operating system, and knowledge correctly before you upgrade Platform to ensure a seamless upgrade experience.

Removing response files generated by previous installations

If you are upgrading from a version before 8.6.0, you must delete the response files that are generated by previous Unica Platform installations. Old response files are not compatible with the 8.6.0 and later installers.

Failure to remove old response files can result in having incorrect data pre-filled in installer fields when the installer is run, or in the installer failing to install some files or skipping configuration steps.

The response file is named `installer.properties`.

The response files for each product are named `installer_productversion.properties`.

The installer creates response files in the directory that you specify during installation. The default location is the home directory of the user.

User account requirement for UNIX™

On UNIX™, the user account that installed the product must complete the upgrade, otherwise the installer fails to detect a previous installation.

32-bit to 64-bit version upgrades

If you are moving from a 32-bit to a 64-bit version of Unica Platform, ensure that you complete the following tasks:

- Ensure that the database client libraries for your product data sources are 64-bit.
- Ensure that all relevant library paths, for example startup or environment scripts, correctly reference the 64-bit versions of your database drivers.

Unloading unused files from memory on AIX®

For installations on AIX®, run the slibclean command that is included with your AIX® installation to unload unused libraries from the memory before you run the installer in the upgrade mode.

> ✏️ **Note:** You must run the slibclean command as a root user.

Backing up custom files

Before starting the upgrade to Unica 12.1.2, backup all the custom files that exist in the `<UNICA_HOME>/jre/` location. If you do not backup the custom files that exist in the `<UNICA_HOME>/jre/` location, you will lose the files because the Unica 12.1.2 upgrade deletes the existing `jre` folder and installs a new `jre` folder containing Oracle JRE files.

> ✏️ **Note:** If the operating system is IBM AIX, Unica 12.1.2 installs IBM JRE.

Auto commit requirement for Oracle or DB2®

If your Unica Platform system tables are in Oracle or DB2®, you must enable the auto-commit mode for the environment open.

See the Oracle or DB2® documentation for instructions.

Change in names of user-defined groups and roles

You must upgrade Unica Platform before you upgrade Unica Campaign. To avoid problems when you upgrade the Unica Platform, the names of the groups and roles that are created by the user must be different from the names of the groups or roles that are defined by Unica Platform.

If the names are same, you must change the group or role names that you created before the upgrade. For example, if you have created a group or role called Admin, you must change the name, because Admin is a name that is used in Unica Campaign.

Time zone support for scheduler

In Unica Platform you can use the Scheduler to configure a process to run at intervals that you define. With the Scheduler, you can schedule Unica Campaign flowchart runs, Unica Optimize sessions and post-optimization flowchart runs, and Deliver mailings.

To take advantage of the time zone support in the Scheduler, edit your scheduled tasks and select the new time zone as required. See the Unica Platform Administrator's Guide for information about using the Scheduler.

Digital Analytics dashboard portlets

If you have any dashboards that contain custom Digital Analytics portlets, you must recreate those portlets after your upgrade is complete.

Backing up the corporatetheme.css file and branding images

If you rebranded the frameset as described in the Unica Platform Administrator's Guide, you must back up the files that you modified before you upgrade Unica Platform. You must restore the files after you complete the upgrade operation, but before you deploy the new version of Unica Platform.

Typically, the `corporatetheme.css` file and branding images have to be modified. The `corporatetheme.css` file and the branding images are located under the `css\theme` directory in the `unica.war` file.

Complete the following steps to back up the `corporatetheme.css` file and the branding images:

1. Make a backup copy of the `unica.war` file before you start the upgrade procedure for Unica Platform.
2. Extract the `unica.war` file and set aside copies of the `corporatetheme.css` file and the branding images.
3. Proceed with the upgrade for Unica Platform, but do not deploy Unica Platform.

4. Extract the new `unica.war` file, and overwrite the existing images and the `corporatetheme.css` file with your backed-up versions.

5. Rewar the new `unica.war` file, and deploy Unica Platform.

See the Unica Platform Administrator's Guide for more details on rebranding.

Unica Platform upgrade scenarios

Upgrade your installation of Unica Platform after you understand the guidelines that apply to your current version of Unica Platform.

Follow the guidelines in the following table for upgrading Unica Platform.

Table 7. Unica Platform upgrade scenarios

Unica Platform source version	Upgrade path
7.x, 8.0.x, 8.1.x, 8.2.x, and 8.5.x	A direct upgrade is not supported. Complete the following steps to upgrade to Unica Platform version 12.1.2. 1. Upgrade from your current version to version 8.6.0. To complete the upgrade to version 8.6.0, obtain the software and follow the instructions in the installation guide for that version. **Note:** Versions before 9.1.1 did not have a separate upgrade guide; upgrade instructions were included in the Installation guide. 2. Upgrade from version 8.6.0 to version 10.0. To complete the upgrade to version 10.0, obtain the software and follow the instructions in the Unica Platform 10.0 Upgrade guide. 3. Upgrade from version 10.0.0 to version 10.1.0.

Table 7. Unica Platform upgrade scenarios (continued)

Unica Platform source version	Upgrade path
	To complete the upgrade to version 10.1, obtain the software and follow the instructions in the Unica Platform 10.1 Upgrade guide. 4. Upgrade from version 10.1.x to version 11.0.0. To complete the upgrade to version 11.0.0, obtain the software and follow the instructions in Unica Platform 11.0.0 Platform upgrade guide. 5. Upgrade from version 11.0.0 to version 11.0.1. To complete the upgrade to version 11.0.1, obtain the software and follow the instructions in this Unica Platform 11.0.1 Upgrade Guide. 6. Upgrade from version 11.0.1 to version 11.1.0. To complete the upgrade to version 11.1.0, obtain the software and follow the instructions in the Unica Platform 11.1.0 Upgrade Guide. 7. Upgrade from version 11.1.0 to version 12.1.0. To complete the upgrade to version 12.1.0, obtain the software and follow the instructions in this Upgrade guide. 8. Upgrade from version 12.1.0 to version 12.1.1. 9. Upgrade from version 12.1.1 to version 12.1.2.
8.6.x and above	Upgrade your installation of Platform as described in one of the following topics. • The Platform upgrade installer can automatically complete the data migration that is required for an upgrade. If you want to allow the installer to create system tables

Table 7. Unica Platform upgrade scenarios (continued)

Unica Platform source version	Upgrade path

	in your database automatically, complete the following steps:
	
	  1. If you are upgrading from a version prior to 11.1, you must first upgrade to version 11.1. For information about upgrading to version 11.1, see the Unica Platform 11.1 Upgrade Guide.
	
	  2. Upgrade to version 11.1. For more information, see Upgrading from version 10.0 and higher with automatic migration *(on page 28)*.
	
	• If your organization's policy does not allow automatic data migration, you must complete the procedure to upgrade manually. If you want to create system tables in your database manually, see one of the following topics.
	
	  ◦ Upgrading from version 8.6.0 with manual migration *(on page 31)*
	  ◦ Upgrading from version 9.0.0 with manual migration *(on page 49)*
	  ◦ Upgrading from version 9.1.0 with manual migration *(on page 65)*
	  ◦ Upgrading from version 9.1.1 with manual migration *(on page 78)*
	  ◦ Upgrading from version 9.1.2 with manual migration *(on page 90)*
	  ◦ Upgrading from version 10.0.0 with manual migration *(on page 101)*
	  ◦ Upgrading from version 10.1.0 with manual migration *(on page 111)*
	  ◦ Upgrading from version 11.0.0 with manual migration *(on page 120)*

Table 7. Unica Platform upgrade scenarios (continued)

Unica Platform source version	Upgrade path
	○ Upgrading from version 11.0.1 with manual migration *(on page 128)* ○ Upgrading from version 11.1.0 with manual migration *(on page 136)* ○ Upgrading from version 12.0.0 with manual migration *(on page 144)* ○ Upgrading from version 12.1.0 with manual migration *(on page 152)* ○ Upgrading from version 12.1.1 with manual migration *(on page 159)*

Correcting the registry file when upgrade installation fails

If installation fails because the installer could not detect the base version of your installed product, you can correct the registry file as described here.

The InstallAnywhere Global registry file, named `.com.zerog.registry.xml`, is created when a Unica product is installed. The registry file tracks all installed Unica products, including their features and components, on that server.

1. Locate the `.com.zerog.registry.xml` file.

   Depending on the server on which you are installing, the `.com.zerog.registry.xml` file is in one of the following locations.

   - On Windows servers, the file is in the `Program Files/Zero G Registry` folder.

     `Zero G Registry` is a hidden directory. You must enable the setting to view hidden files and folders.

   - On UNIX systems, the file is in one of the following directories.

- Root user - `/var/`
- Non-root user - `$HOME/`

2. Make a backup copy of the file.

3. Edit the file to change all entries that refer to the version of your installed product.

   For example, this is a section of the file that corresponds to Unica Plan version 8.6.0.3.

   ```
   <product name="Plan" id="dd6f88e0-1ef1-11b2-accf-c518be47c366"

   version=" 8.6.0.3 " copyright="2013" info_url="" support_url=""

   location="<HCL_Unica_Home>\Plan" last_modified="2013-07-25 15:34:01">
   ```

   In this case, you would change all entries that refer to `version=" 8.6.0.3 "` to the base version, which is 8.6.0.0 in this case.

Chapter 3. Upgrading from version 10.0 and higher with automatic migration

Automatic migration allows the installer to run the SQL script and insert the configuration properties that are required to upgrade your Unica Platform system tables. If your company policy does not allow automated updates to your system table database, you must use manual migration, described elsewhere in this guide.

> ⚠️ **Important:** If you are upgrading from a version prior to 10.1, you must first upgrade to version 10.1 and upgrade to version 11.0.1 and then upgrade to version 11.1. For information about upgrading to version 10.1, 11.0.1, and 11.1 see the IBM Marketing Platform 10.1 Upgrade Guide and the IBM Marketing Platform 11.0.1 Upgrade Guide.
>
> The supported upgrade path via installer is 11.0.1 to 11.1.0. Users on version 11.0 can move to 11.0.1 by doing manual upgrade or fast upgrade only.

Ensure that you have the following installers in the same directory.

- The Unica master installer
- The Unica Platform installer

Use the following guidelines as a best practice.

- Place the installers in the same directory where you originally placed the installers for the earlier versions of your products.
- Remove any earlier versions of the Unica product installers from the directory to avoid having the master installer attempt to install the earlier versions.

Upgrading from version 11.1 and higher is an in-place upgrade. In an in-place upgrade, you install the upgraded version of Unica Platform to the directory where your current Unica Platform is installed.

See for information about upgrading from earlier versions.

1. Create a back-up copy of your Unica Platform system table database.

   > ⚠️ **Important:** Do not skip this step. If the upgrade operation fails, you cannot roll back your database and your data is corrupted.

2. Undeploy your Unica Platform deployment.
3. Run the Unica master installer.
   The Unica master installer starts. Use the following instructions after theUnica master installer starts.
      - When the Unica master installer prompts you to choose an installation directory, choose the root installation directory, not the Unica Platform installation directory, which is under this root directory.
      - When the Unica master installer prompts you to enter Unica Platform database connection information, enter the information that pertains to your current Unica Platform system tables.

   The Unica master installer pauses and launches the Unica Platform installer.
4. Complete the following steps while the Unica Platform installer is running.

   a. When the Unica Platform installer prompts you for an installation directory, select the directory of your current Unica Platform installation, usually named `Platform`.

   b. Select **Automatic database setup**.

   c. If Contact Central is selected in component selection during installation, select database type.

      > ✏️ **Note:** Contact Central is supported with OneDb and Oracle database.

       d. Provide Contact Central database connection details. Notes. By default only automatic database setup will be executed for Control Center.

       e. Follow the remaining steps in the installation wizard and enter the requested information.

5. Use the quartzjobtool to update scheduler jobs. This is a required step. If this upgrade tool is not run, any existing scheduled job fails 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

6. Deploy your installation.

> **Note:** Pay close attention to the installation summary windows. If errors are reported, check the installer log files and contact Unica technical support if necessary.

Chapter 4. Upgrading from version 8.6.0 with manual migration

The Unica Platform upgrade installer can perform all of the data migration that is required for an upgrade automatically. However, if the policy of your organization does not allow automatic migration, you must complete the migration procedure to upgrade Unica Platform manually.

Ensure that you have the following installers in the same directory.

- The master installer
- The Unica Platform installer

To upgrade to Unica Platform version 12.1.2 manually, you must run SQL scripts and run some command line utilities to populate the system tables. Ensure that your installation of Unica Platform is fully functional so that you can run the utilities. Complete information about using these utilities, including example commands for common tasks, is available in the following topics.

- populateDb *(on page 198)*
- configTool *(on page 184)*
- alertConfigTool *(on page 183)*

The utilities are in the `tools\bin` directory under your Unica Platform installation.

See Unica Platform upgrade scenarios (on page 23) for information about upgrading from other versions.

1. Create a back-up copy of your Unica Platform system table database.

    ⚠️ **Important:** Do not skip this step. If the upgrade operation fails, you cannot roll back your database and your data is corrupted.

2. Undeploy your Unica Platform deployment.
3. Run the Unica master installer.

The Unica master installer starts. Use the following instructions after theUnica master installer starts.

- When the Unica master installer prompts you to choose an installation directory, choose the root installation directory, not the Unica Platform installation directory that is under this root directory.
- When the Unica master installer prompts you to enter Unica Platform database connection information, enter the information that pertains to your current Unica Platform system tables.

The Unica master installer pauses and starts the Unica Platform installer.

4. Complete the following steps while the Unica Platform installer is running.

   a. When the Unica Platform installer prompts you for an installation directory, select the directory of your current Unica Platform installation, usually named `Platform`.

   b. Allow the installer to create a back-up copy of your previous installation of Unica Platform.

   c. If Contact Central is selected in component selection during installation, select database type.

   > **Note:** Contact Central is supported with OneDb and Oracle database.

   d. Provide Contact Central database connection details. Notes. By default only automatic database setup will be executed for Control Center.

   e. Select **Manual database setup**.

   f. Clear the **Run Platform configuration** check box.

   g. Follow the remaining steps in the installation wizard and enter the requested information.

5. Run the following scripts against your system tables, where *DB_Type* is the database type.

Table 8. SQL scripts for upgrades from version 8.6.0

File details	File location
db2_unicode_fix_90.sql Applies only to DB2 system tables	db\upgrade86to90 directory under your Unica Platform installation
drop-liferay-tables.sql	db\upgrade86to90 directory under your Unica Platform installation
ManagerSchema_*DB_Type*_90upg.sql	db\upgrade86to90 directory under your Unica Platform installation
ManagerSchema_*DB_Type*_91upg.sql	db\upgrade90to91 directory under your Unica Platform installation
ManagerSchema_*DB_Type*_911upg.sql	db\upgrade91to911 directory under your Unica Platform installation
ManagerSchema_*DB_Type*_10upg.sql	db\upgrade912to10 directory under your Unica Platform installation
DB_Type_QRTZ_Scheduler_10_upgrade_-Script.sql	db\upgrade912to10 directory under your Unica Platform installation
ManagerSchema_DB_Type_10002upg.sql	db\upgrade10001to10002 directory under your Unica Platform installation
ManagerSchema_DB_Type_101upg.sql	db\upgrade10002to101 directory under your Unica Platform installation
ManagerSchema_*DB_Type*_11upg.sql	db\upgrade101to11 directory under your Unica Platform installation
ManagerSchema_*DB_Type*_111upg.sql	db\upgrade1101to111 directory under your Unica Platform installation

File details	File location
`ManagerSchema_DB_Type_11102upg.sql`	`db\upgrade1111to11102` directory under your Unica Platform installation
`ManagerSchema_DB_Type_12upg.sql`	`db\upgrade1102to12` directory under your Unica Platform installation
`ManagerSchema_DB_Type_121upg.sql`	`db\upgrade12to121` directory under your Unica Platform installation
`ManagerUpdate_12101upg.sql`	`db\upgrade121to12101` directory under your Unica Platform installation
`ManagerUpdate_12103_DB_Type_upg.sql`	`db\upgrade12101to12103` directory under your Unica Platform installation
`ManagerUpdate_12104_DB_Type_upg.sql` and `DB_Type_QRTZ_Scheduler_12104_upgrade_Script.sql`	`db\upgrade12103to12104` directory under your Unica Platform installation
`ManagerUpdate_1211_DB_Type_upg.sql`	`db\upgrade12104to1211` directory under your Unica Platform installation
`ManagerUpdate_1212_DB_Type_upg.sql` and `ContentIntegrationUpdate_1212_DB_Type_upg.sql`	`db\upgrade1211to1212` directory under your Unica Platform installation

6. Run the `upgrade86to90` batch or shell script, located the `tools\bin\upgrade86to90` directory under your Unica Platform installation.

7. Run the `upgrade90to91` batch or shell script, located the `tools\bin\upgrade90to91` directory under your Unica Platform installation.

8. Use the `populateDb` utility to populate the system tables with default Unica Platform configuration properties, users and groups, and security roles and permissions.

   The `populateDb` utility is in the `tools\bin` directory under your Unica Platform installation. Example command: `populateDb -n Manager`

9. Use the configTool utility to import configuration properties as described below.

⚠ **Important:** Perform the imports in the order shown in the following table.

Table 9. Configuration properties for upgrades from version 8.6.0

File details	Example command
Skip this import if the property named **LDAP BaseDN periodic search enabled** is present under the **Platform** | **Security** | **Login method details** | **LDAP synchronization** category. If the property is not present, perform the following import. • File: `Ldap_Auto_Sync_BaseDN_Settings.xml` • Location: `conf\upgrade86to90` directory under your Unica Platform installation • Purpose: Import a configuration property that enables LDAP import search by DN	configTool.bat -i -p "Affinium|suite|security| ldapSynchronization|ldapProperties" -f "C:\Unica\Platform\conf \upgrade86to90\Ldap_Auto_Sync_BaseDN_Settings.xml"
• File: `quicklinks_category.xml` • Location: `conf\upgrade86to90` directory under your Unica Platform installation • Purpose: Import properties for quicklinks dashboard portlets	configTool.bat -i -o -p "Affinium|suite" -f "C:\Unica\Platform\conf\upgrade86to90\quicklinks_category.xml"

File details	Example command
• File: `communication_e-mail.xml` • Location: `conf\upgrade86to90` directory under your Unica Platform installation • Purpose: Import configuration properties to enable email notifications	configTool.bat -i -o -p "Affinium|Manager" -f "C:\Unica\Platform\conf\upgrade86to90\communication_email.xml"

• File: `notification.xml`
• Location: `conf\upgrade86to90` directory under your Unica Platform installation
• Purpose: Import configuration properties for the notification feature. The following properties are added. Default values are as follows.
   ◦ The number of days to retain alerts - 90
   ◦ Frequency in minutes at which emails are sent - 30
   ◦ Maximum number of retries for sending an email - 1
	configTool.bat -i -o -p "Affinium|suite" -f "C:\Unica\Platform\conf\upgrade86to90\notification.xml"

• File: `manager_alerts_registration.xml` • Location: `conf` directory under your Unica Platform installation • Purpose: Import configuration properties that create the alerts menu items	configTool.bat -i -o -p "Affinium|suite|uiNavigation|alerts" -f "C:\Unica\Platform\conf\manager_alerts_registration.xml"

File details	Example command
• File: `is_clustered.xml` • Location: `conf\upgrade90to91` directory under your Unica Platform installation • Purpose: The boolean property to specify if the Unica Platform instance is clustered. If you install Unica Platform in a clustered deployment, set this property to True. Otherwise, retain the default value of False.	configTool.bat -i -p "Affinium|suite" -f "C:\Unica\Platform\conf\up-grade90to91\is_clustered.xml"
• File: `taskNotificationNavIm-port.xml` • Location: `conf\upgrade90to91` directory under your Unica Platform installation • Purpose: Import the **My job notifications** navigation option under the **Settings** menu	configTool.bat -i -p "Affinium|suite|uiNav-igation|settingsMenu" -f "C:\Unica\Plat-form\conf\upgrade90to91\taskNotifica-tionNavImport.xml"
• File: `taskNotification.xml` • Location: `conf\upgrade90to91` directory under your Unica Platform installation • Purpose: Imports the **Group Name** field where you can enter the groups that are notified with scheduled notifications	configTool.bat -i -p "Affinium|suite|sched-uler|taskRegistrations|Campaign|flow-chart" -f "C:\Unica\Platform\conf\up-grade90to91\taskNotification.xml" configTool.bat -i -p "Affinium|suite|sched-uler|taskRegistrations|Campaign|mail-ing" -f "C:\Unica\Platform\conf\up-grade90to91\taskNotification.xml" configTool.bat -i -p "Affinium|suite|sched-uler|taskRegistrations|Campaign|opti-

File details	Example command
	mize" -f "C:\Unica\Platform\conf\up-grade90to91\taskNotification.xml" configTool.bat -i -p "Affinium|suite|sched-uler|taskRegistrations|InteractionHisto-ry|ETL" -f "C:\Unica\Platform\conf\up-grade90to91\taskNotification.xml" configTool.bat -i -p "Affinium|suite|scheduler|taskRegistrations|Attribution-Modeler|TrainingRun" -f "C:\Unica\Plat-form\conf\upgrade90to91\taskNotifica-tion.xml" configTool.bat -i -p "Affinium|suite|scheduler|taskRegistrations|Attribution-Modeler|ScoringRun" -f "C:\Unica\Plat-form\conf\upgrade90to91\taskNotifica-tion.xml"
• File: `cognos.xml` • Location: `conf\upgrade90to91` directory under your Unica Platform installation • Purpose: Update the Cognos version	configTool.bat -i -o -p "Affinium|Report|in-tegrations|cognos10" -f "C:\Unica\Plat-form\conf\upgrade90to91\cognos.xml"
• File: `scheduler.xml` • Location: `conf\upgrade90to91` directory under your Unica Platform installation • Purpose: The boolean property to enable or disable the scheduler feature in Unica Platform	configTool.bat -i -p "Affinium|suite|sched-uler" -f "C:\Unica\Platform\conf\up-grade90to91\scheduler.xml"

File details	Example command
• File: `unknownPolling.xml` • Location: `conf\upgrade90to91` directory under your Unica Platform installation • Purpose: Imports properties for quicklinks dashboard portlets	configTool.bat -i -p "Affinium|suite|scheduler" -f "C:\Unica\Platform\conf\upgrade90to91\unknownPolling.xml"
• File: `config_data_filter_-cache.xml` • Location: `conf\upgrade90to91` directory under your Unica Platform installation • Purpose: Allows you to configure the data filter cache. If the configuration property is set to True, the data filter cache is enabled for enhanced performance. If the configuration property is set to False, the caching is not enabled, and all operations are completed by connecting to the database for each data filter request. The configuration property is visible on the user interface.	configTool.bat -vp -p "Affinium|Manager|datafiltering" -f "C:\Unica\Platform\conf\upgrade90to91\Config_data_filter_cache.xml" ✏️ **Note:** If you change this value, you must restart the Unica Platform for the changes to take effect.
• File: `Refresh_data_filter_-cache.xml` • Location: `conf\upgrade90to91` directory under your Unica Platform installation • Purpose: If data filter caching is enabled, this configuration property	configTool.bat -vp -p "Affinium|Manager|datafiltering" -f "C:\Unica\Platform\conf\upgrade90to91\Refresh_data_filter_cache.xml"

File details	Example command
ensures that the cache is refreshed every time that there is a change in the data filter definitions	
• Files: `emm_spss_navigation.xml` and `emm_spss_configuration.xml` • Location: `conf` directory under your Unica Platform installation • Purpose: Import IBM SPSS Modeler Advantage Enterprise Marketing Management Edition configuration properties	configTool.bat -i -o -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f "C:\Unica\Platform\conf\emm_spss_navigation.xml" configTool.bat -i -o -p "Affinium" -f "C:\Unica\Platform\conf\emm_spss_configuration.xml"
• File: `AuditEvents.xml` • Location: `conf\upgrade91to911` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Audit Events** configuration node	configTool.bat -i -p "Affinium|suite" -f C:\Unica\Platform\conf\upgrade91to911\AuditEvents.xml
• File: `FederatedAuthentication.xml` • Location: `conf\upgrade91to911` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | Federated Authentication** configuration node	configTool.bat -i -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade91to911\ FederatedAuthentication.xml

File details	Example command
• File: `MO_bulk_deactivation_-scheduler.xml` • Location: `conf\upgrade91to911` directory under your Unica Platform installation • Purpose: Import the configuration node for the bulk deactivation schedule feature for Unica Plan	configTool.bat -i -p "Affinium|suite|scheduler| taskRegistrations" -f C:\Unica\Platform\conf\upgrade91to911\ MO_bulk_deactivation_scheduler.xml
• File: `emm_audit_navigation.xml` • Location: `conf` directory under your Unica Platform installation • Purpose: Import the **Analytics > Platform** menu item for Audit Event reports	configTool.bat -i -p "Affinium|suite|uiNavigation|mainMenu| Analytics" -f C:\Unica\Platform\conf\ emm_audit_navigation.xml
• File: `APISecurity.xml` • Location: `conf\upgrade911to912` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API management** configuration node	configTool.bat -i -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade911to912\APISecurity.xml
• File: `APISecurity.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API management** configuration node	configTool.bat -vp -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade912to10\APISecurity.xml Note that this version of the `APISecurity.xml` file is different from the one shown in the previous row and is in a different location.

File details	Example command
• File: `SAMLAuthentication.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | SAML2.0** configuration node	configTool.bat -vp -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade912to10\SAMLAuthentication.xml
• File: `SAMLAuthenticationDetails.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | Login method details | SAML 2.0** configuration nodes	configTool.bat -vp -p "Affinium|suite|security|loginModes" -f C:\Unica\Platform\conf\upgrade912to10\SAMLAuthenticationDetails.xml
• File: `ExternalScheduler.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Scheduler | Scheduler registrations | Unica Platform** configuration node	configTool.bat -vp -p "Affinium|suite|scheduler| taskRegistrations" -f C:\Unica\Platform\conf\upgrade912to10\ExternalScheduler.xml

File details	Example command
• File: `JWTAuthentication.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | JWT authentication** configuration node	configTool.bat -vp -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade912to10\JWTAuthentication.xml
• File: `SecureSuiteStaticContent.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Apply security on static content for all applications** property under the **Unica Platform** configuration node	configTool.bat -vp -p "Affinium|suite" -f C:\Unica\Platform\conf\upgrade912to10\SecureSuiteStaticContent.xml
• File: `APISecurity_interactCollection.xml` • Location: `conf\upgrade10to1001` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API Management | Unica Campaign** configuration node	configTool.bat -vp -p "Affinium|suite|security|apiSecurity|campaign" -f <Platform_Home>\conf\upgrade10to1001\APISecurity_interactCollection.xml

File details	Example command
• File: `APISecurity_triggered-Messages.xml` • Location: `conf\upgrade10to1001` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API Management | Unica Campaign** configuration node	configTool.bat -vp -p "Affinium|suite|security|apiSecurity|campaign" -f <Platform_Home>\conf\upgrade10to1001\APISecurity_triggered-Messages.xml
• File: `supportServer_config.xml` • Location: `conf\upgrade10to101` directory under your Unica Platform installation • Purpose: Import the **HCL Unica** configuration nodes	configTool.bat -vp -p "Affinium" -f <Platform_Home>\conf\upgrade10to101\supportServer_config.xml
• File: `cognos11_integration.xml` • Location: `conf\upgrade101to11` directory under your Unica Platform installation • Purpose: Import the **Affinium|Report|integrations|cognos11** configuration node	configTool.bat -vp -p "Affinium|Report|integrations" -f <Platform_Home>/conf/upgrade101to11/cognos11_integration.xml
• File: watsonAssistant_navigation.xml • Location: `<PLATFORM_HOME>\conf\watsonAssistant_navigation.xml` directory under your Unica Platform installation	configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f "<PLATFORM_HOME>\conf\watsonAssistant_navigation.xml"

File details	Example command
• Purpose: Import the Affinium|suite|uiNavigation|mainMenu|Analytics|Watson Assistantconfiguration node configuration node	
• File: APISecurity.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the `Unica Platform | Security | API Management | Unica Platform | License node`	`configTool -vp -p "Affinium|suite|security" -f "<PLATFORM_HOME>\conf\upgrade12to121\APISecurity.xml"`
• APISecurity_Deliver.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the `Unica Platform | Security | API Management | Unica Deliver configuration node`	`configTool.bat -v -i -p "Affinium|suite|security|apiSecurity" -f "<PLATFORM_HOME>\conf\upgrade12to121\APISecurity_Deliver.xml"`
• LicenseDetailsNavigation.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the License details navigation option under the Settings menu	`configTool.bat -vp -p "Affinium| suite|uiNavigation|settingsMenu" -f "<PLATFORM_HOME>\conf\upgrade12to121\LicenseDeatilsNavigation.xml"`
File: APISecurity_GetInstalledApps.xml Location: `<PLATFORM_HOME>\conf\upgrade121to12103` Purpose: Import new API configuration	`configTool.bat -vp -p "Affinium|suite|security|apiSecurity|manager" -f APISecurity_GetInstalledApps.xml`

File details	Example command

File: PlanJourneyIntegrationOption.xml
 	
Location: `<PLATFORM_HOME>\conf`
`\upgrade121to12103`
 	
Purpose: Import Plan Journey integration property
	`configTool.bat -vp -p "Affini-`
	`um|suite" -f PlanJourneyIntegra-`
	`tionOption.xml`

File: `SiteMinder_AdditionalHeader-`
`Variables.xml`
 	
Location: `<PLATFORM_HOME>\conf`
`\upgrade121to12103`
 	
Purpose: Import new property to ac-cept additional header variables for Site-minder
	`configTool.bat -vp -p "Affini-`
	`um|suite|security|loginModes|site-`
	`MinderPartitionLogin" -f Site-`
	`Minder_AdditionalHeaderVari-`
	`ables.xml`

File: `PasswordRules.xml` Location: `<PLATFORM_HOME>\conf` `\upgrade12103to12104` Purpose: Import new password rules	`configTool.bat -vp -p "Affini-` `um|Manager" -f PasswordRules.xml`

File: `support_Collaborate.xml`
 	
Location: `<PLATFORM_HOME>\conf`
`\upgrade12103to12104`
 	
Purpose: Update CollaborateSupport-Server property with new value
	`configTool.bat -i -o -p "Affini-`
	`um|suite" -f support_Collabo-`
	`rate.xml`

File: `LookerIntegration.xml`
 	
Location: `<PLATFORM_HOME>\conf`
`\upgrade12104to1211`
 	
Purpose: Import Google Looker integra-tion configuration
	`configtool.bat -vp -p "Affinium|Re-`
	`port|integrations" -f LookerInte-`
	`gration.xml`

File details	Example command

File: `OpenInsights_navigation.xml`
 	
Location: `<PLATFORM_HOME>\conf`
`\upgrade12104to1211`
 	
Purpose: Import navigation configuration for Open Insights menu
	`configtool.bat -vp -p "Affini-`
	`um|suite|uiNavigation|mainMenu|An-`
	`alytics" -f OpenInsights_naviga-`
	`tion.xml`

File: `OpenInsightsSeparator_naviga-`
`tion.xml`
 	
Location: `<PLATFORM_HOME>\conf`
`\upgrade12104to1211`
 	
Purpose: Import configuration for adding separator for Open Insights menu
	`configtool.bat -vp -p "Affini-`
	`um|suite|uiNavigation|mainMenu|An-`
	`alytics" -f OpenInsightsSeparator_-`
	`navigation.xml`

10. Use the `alertConfigTool` utility to register the Platform alerts and notifications, as follows.

    The `alertConfigTool` utility is in the `tools\bin` directory under your Unica Platform installation.

    Run this utility from the `tools\bin` directory. You reference the `Platform_alerts_configuration.xml` file, which is in the `conf` directory under your Unica Platform installation.

    Example command (Windows): `alertConfigTool.bat -i -f C:\Unica\Platform` `\conf\Platform_alerts_configuration.xml`

11. Restart the web application server on which Unica Platform is deployed so the changes are applied.

12. Complete the following steps to update the **Help > About** page.

    a. Use the **configTool** utility to export the **Affinium | Manager | about** category.

       ✏️ **Note:** The **Affinium | Manager | about** category is not visible on the **Configuration** page because it is marked hidden.

Example (Windows™):

```
configTool.bat -x -p "Affinium|Manager|about" -f "C:\Unica\Platform
\conf\about.xml"
```

b. Edit the exported XML file that you just created (`about.xml` in the example) to change the version number and display name.

Find the `releaseNumber` property and change the value to the current version of Unica Platform. Change the value of the `copyright` property to the appropriate copyright year.

In the following example, change the release number from 8.6.0.0.0 to 12.1.2.0.0 and change the copyright to 2021.

```
<property name="releaseNumber" type="string">
<displayNameKey>about.releaseNumber</displayNameKey>
<value>8.6.0.0.0.build_number</value>
</property>
<property id="541" name="copyright" type="string_property"
 width="40">
<value>2016</value>
</property>
```

c. Use the **configTool** utility to import the revised file.

You must use the `-o` option to overwrite the node. You must specify the parent node when you import.

Example (Windows™):

```
configTool.bat -vp -i -p "Affinium|Manager" -f "C:\Unica\Platform
\conf\about.xml" -o
```

13. Deploy and verify your installation as described in the

Chapter 5. Upgrading from version 9.0.0 with manual migration

The Unica Platform upgrade installer can perform all of the data migration that is required for an upgrade automatically. However, if the policy of your organization does not allow automatic migration, you must complete the migration procedure to upgrade Unica Platform manually.

Ensure that you have the following installers in the same directory.

- The master installer
- The Unica Platform installer

To upgrade to Unica Platform version 12.1.2 manually, you must run SQL scripts and run some command line utilities to populate the system tables. Ensure that your installation of Unica Platform is fully functional so that you can run the utilities. Complete information about using these utilities, including example commands for common tasks, is available in the following topics.

- populateDb *(on page 198)*
- configTool *(on page 184)*

The utilities are in the `tools\bin` directory under your Unica Platform installation.

See Unica Platform upgrade scenarios (on page 23) for information about upgrading from other versions.

1. Create a back-up copy of your Unica Platform system table database.

   ⚠ **Important:** Do not skip this step. If the upgrade operation fails, you cannot roll back your database and your data is corrupted.

2. Undeploy your Unica Platform deployment.
3. Run the Unica master installer.

The Unica master installer starts. Use the following instructions after theUnica master installer starts.

- When the Unica master installer prompts you to choose an installation directory, choose the root installation directory, not the Unica Platform installation directory that is under this root directory.
- When the Unica master installer prompts you to enter Unica Platform database connection information, enter the information that pertains to your current Unica Platform system tables.

The Unica master installer pauses and starts the Unica Platform installer.

4. Complete the following steps while the Unica Platform installer is running.

   a. When the Unica Platform installer prompts you for an installation directory, select the directory of your current Unica Platform installation, usually named `Platform`.

   b. Allow the installer to create a back-up copy of your previous installation of Unica Platform.

   c. If Contact Central is selected in component selection during installation, select database type.

      > ✏️ **Note:** Contact Central is supported with OneDb and Oracle database.

   d. Provide Contact Central database connection details. Notes. By default only automatic database setup will be executed for Control Center.

   e. Select **Manual database setup**.

   f. Clear the **Run Platform configuration** check box.

   g. Follow the remaining steps in the installation wizard and enter the requested information.

5. Run the following scripts against your system tables, where *DB_Type* is the database type.

Table 10. SQL scripts for upgrades from version 9.0

File details	File location
`ManagerSchema_DB_Type_91upg.sql`	`db\upgrade90to91` directory under your Unica Platform installation
`ManagerSchema_DB_Type_911upg.sql`	`db\upgrade91to911` directory under your Unica Platform installation
`ManagerSchema_DB_Type_10upg.sql`	`db\upgrade912to10` directory under your Unica Platform installation
`DB_Type_QRTZ_Scheduler_10_upgrade_-Script.sql`	`db\upgrade912to10` directory under your Unica Platform installation
`ManagerSchema_DB_Type_10002upg.sql`	`db\upgrade10001to10002` directory under your Unica Platform installation
`ManagerSchema_DB_Type_101upg.sql`	`db\upgrade10002to101` directory under your Unica Platform installation
`ManagerSchema_DB_Type_11upg.sql`	`db\upgrade101to11` directory under your Unica Platform installation
`ManagerSchema_DB_Type_111upg.sql`	`db\upgrade1101to111` directory under your Unica Platform installation
`ManagerSchema_DB_Type_11102upg.sql`	`db\upgrade1111to11102` directory under your Unica Platform installation
`ManagerSchema_DB_Type_12upg.sql`	`db\upgrade1102to12` directory under your Unica Platform installation
`ManagerSchema_DB_Type_121upg.sql`	`db\upgrade12to121` directory under your Unica Platform installation
`ManagerUpdate_12101upg.sql`	`db\upgrade121to12101` directory under your Unica Platform installation

File details	File location
`ManagerUpdate_12103_DB_Type_upg.sql`	`db\upgrade12101to12103` directory under your Unica Platform installation
`ManagerUpdate_12104_DB_Type_upg.sql` and `DB_Type_QRTZ_Scheduler_12104_-upgrade_Script.sql`	`db\upgrade12103to12104` directory under your Unica Platform installation
`ManagerUpdate_1211_DB_Type_upg.sql`	`db\upgrade12104to1211` directory under your Unica Platform installation
`ManagerUpdate_1212_DB_Type_upg.sql` and `ContentIntegrationUpdate_1212_-DB_Type_upg.sql`	`db\upgrade1211to1212` directory under your Unica Platform installation

6. Run the `upgrade90to91` batch or shell script, located the `tools\bin\upgrade90to91` directory under your Unica Platform installation.

7. Use the `populateDb` utility to populate the system tables with default Unica Platform configuration properties, users and groups, and security roles and permissions.

   The `populateDb` utility is in the `tools\bin` directory under your Unica Platform installation. Example command: `populateDb -n Manager`

8. Use the configTool utility to import configuration properties as described below.

> ❗ **Important:** Perform the imports in the order shown in the following table.

Table 11. Configuration properties for upgrades from version 9.0.0

File details	Example command
• File: `is_clustered.xml` • Location: `conf\upgrade90to91` directory under your Unica Platform installation • Purpose: The boolean property to specify if the Unica Platform instance is clustered. If you install	configTool.bat -i -p "Affinium|suite" -f "C:\Unica\Platform\conf\up-grade90to91\is_clustered.xml"

File details	Example command
Unica Platform in a clustered deployment, set this property to True. Otherwise, retain the default value of False.	
• File: `taskNotificationNavImport.xml` • Location: `conf\upgrade90to91` directory under your Unica Platform installation • Purpose: Import the **My job notifications** navigation option under the **Settings** menu	configTool.bat -i -p "Affinium|suite|uiNavigation|settingsMenu" -f "C:\Unica\Platform\conf\upgrade90to91\taskNotificationNavImport.xml"
• File: `taskNotification.xml` • Location: `conf\upgrade90to91` directory under your Unica Platform installation • Purpose: Imports the **Group Name** field where you can enter the groups that are notified with scheduled notifications	configTool.bat -i -p "Affinium|suite|scheduler|taskRegistrations|Campaign|flowchart" -f "C:\Unica\Platform\conf\upgrade90to91\taskNotification.xml" configTool.bat -i -p "Affinium|suite|scheduler|taskRegistrations|Campaign|mailing" -f "C:\Unica\Platform\conf\upgrade90to91\taskNotification.xml" configTool.bat -i -p "Affinium|suite|scheduler|taskRegistrations|Campaign|optimize" -f "C:\Unica\Platform\conf\upgrade90to91\taskNotification.xml" configTool.bat -i -p "Affinium|suite|scheduler|taskRegistrations|InteractionHistory|ETL" -f "C:\Unica\Platform\conf\upgrade90to91\taskNotification.xml"

File details	Example command
	configTool.bat -i -p "Affinium|suite|scheduler|taskRegistrations|Attribution-Modeler|TrainingRun" -f "C:\Unica\Platform\conf\upgrade90to91\taskNotification.xml" configTool.bat -i -p "Affinium|suite|scheduler|taskRegistrations|Attribution-Modeler|ScoringRun" -f "C:\Unica\Platform\conf\upgrade90to91\taskNotification.xml"
• File: `cognos.xml` • Location: `conf\upgrade90to91` directory under your Unica Platform installation • Purpose: Update the Cognos version	configTool.bat -i -o -p "Affinium|Report|integrations|cognos10" -f "C:\Unica\Platform\conf\upgrade90to91\cognos.xml"
• File: `scheduler.xml` • Location: `conf\upgrade90to91` directory under your Unica Platform installation • Purpose: The boolean property to enable or disable the scheduler feature in Unica Platform	configTool.bat -i -p "Affinium|suite|scheduler" -f "C:\Unica\Platform\conf\upgrade90to91\scheduler.xml"
• File: `unknownPolling.xml` • Location: `conf\upgrade90to91` directory under your Unica Platform installation • Purpose: Imports properties for quicklinks dashboard portlets	configTool.bat -i -p "Affinium|suite|scheduler" -f "C:\Unica\Platform\conf\upgrade90to91\unknownPolling.xml"

File details	Example command
• File: `config_data_filter_-cache.xml` • Location: `conf\upgrade90to91` directory under your Unica Platform installation • Purpose: Allows you to configure the data filter cache. If the configuration property is set to True, the data filter cache is enabled for enhanced performance. If the configuration property is set to False, the caching is not enabled, and all operations are completed by connecting to the database for each data filter request. The configuration property is visible on the user interface.	configTool.bat -vp -p "Affinium|Manager|datafiltering" -f "C:\Unica\Platform\conf\upgrade90to91\Config_data_filter_cache.xml" 📝 **Note:** If you change this value, you must restart the Unica Platform for the changes to take effect.
• File: `Refresh_data_filter_-cache.xml` • Location: `conf\upgrade90to91` directory under your Unica Platform installation • Purpose: If data filter caching is enabled, this configuration property ensures that the cache is refreshed every time that there is a change in the data filter definitions	configTool.bat -vp -p "Affinium|Manager|datafiltering" -f "C:\Unica\Platform\conf\upgrade90to91\Refresh_data_filter_cache.xml"

File details	Example command
• Files: `emm_spss_navigation.xml` and `emm_spss_configuration.xml` • Location: `conf` directory under your Unica Platform installation • Purpose: Import IBM SPSS Modeler Advantage Enterprise Marketing Management Edition configuration properties	configTool.bat -i -o -p "Affinium|suite|ui-Navigation|mainMenu|Analytics" -f "C:\Unica\Platform\conf\emm_spss_navigation.xml" configTool.bat -i -o -p "Affinium" -f "C:\Unica\Platform\conf\emm_spss_configuration.xml"
• File: `AuditEvents.xml` • Location: `conf\upgrade91to911` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Audit Events** configuration node	configTool.bat -i -p "Affinium|suite" -f C:\Unica\Platform\conf\upgrade91to911\AuditEvents.xml
• File: `FederatedAuthentication.xml` • Location: `conf\upgrade91to911` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | Federated Authentication** configuration node	configTool.bat -i -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade91to911\ FederatedAuthentication.xml

File details	Example command
• File: `MO_bulk_deactivation_-scheduler.xml` • Location: `conf\upgrade91to911` directory under your Unica Platform installation • Purpose: Import the configuration node for the bulk deactivation schedule feature for Unica Plan	configTool.bat -i -p "Affinium|suite|scheduler| taskRegistrations" -f C:\Unica\Platform\conf\upgrade91to911\MO_bulk_deactivation_scheduler.xml
• File: `emm_audit_navigation.xml` • Location: `conf` directory under your Unica Platform installation • Purpose: Import the **Analytics > Platform** menu item for Audit Event reports	configTool.bat -i -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f C:\Unica\Platform\conf\emm_audit_navigation.xml
• File: `APISecurity.xml` • Location: `conf\upgrade911to912` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API management** configuration node	```configTool.bat -i -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade911to912\APISecurity.xml```
• File: `APISecurity.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API management** configuration node	configTool.bat -vp -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade912to10\APISecurity.xml Note that this version of the `APISecurity.xml` file is different from the one shown in the previous row and is in a different location.

File details	Example command
• File: `SAMLAuthentication.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | SAML2.0** configuration node	configTool.bat -vp -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade912to10\SAMLAuthentication.xml
• File: `SAMLAuthenticationDetails.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | Login method details | SAML 2.0** configuration nodes	configTool.bat -vp -p "Affinium|suite|security|loginModes" -f C:\Unica\Platform\conf\upgrade912to10\SAMLAuthenticationDetails.xml
• File: `ExternalScheduler.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Scheduler | Scheduler registrations | Unica Platform** configuration node	configTool.bat -vp -p "Affinium|suite|scheduler| taskRegistrations" -f C:\Unica\Platform\conf\upgrade912to10\ExternalScheduler.xml

File details	Example command
• File: `JWTAuthentication.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | JWT authentication** configuration node	configTool.bat -vp -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade912to10\JWTAuthentication.xml
• File: `SecureSuiteStaticContent.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Apply security on static content for all applications** property under the **Unica Platform** configuration node	configTool.bat -vp -p "Affinium|suite" -f C:\Unica\Platform\conf\upgrade912to10\SecureSuiteStaticContent.xml
• File: `APISecurity_interactCollection.xml` • Location: `conf\upgrade10to1001` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API Management | Unica Campaign** configuration node	configTool.bat -vp -p "Affinium|suite|security|apiSecurity|campaign" -f <Platform_Home>\conf\upgrade10to1001\APISecurity_interactCollection.xml

File details	Example command
• File: `APISecurity_triggeredMes-sages.xml` • Location: `conf\upgrade10to1001` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API Management | Unica Campaign** configuration node	configTool.bat -vp -p "Affinium|suite|security|apiSecurity|campaign" -f <Platform_Home>\conf\up-grade10to1001\APISecurity_triggered-Messages.xml
• File: `supportServer_config.xml` • Location: `conf\upgrade10to101` directory under your Unica Platform installation • Purpose: Import the **HCl Unica** con-figuration nodes	configTool.bat -vp -p "Affinium" -f <Plat-form_Home>\conf\upgrade10to101\sup-portServer_config.xml
• File: `cognos11_integration.xml` • Location: `conf\upgrade101to11` directory under your Unica Platform installation • Purpose: Import the **Affinium|Re-port|integrations|cognos11** config-uration node	configTool.bat -vp -p "Affinium|Report|in-tegrations" -f <Platform_Home>/conf/up-grade101to11/cognos11_integration.xml
• File: watsonAssistant_naviga-tion.xml • Location: `<PLATFORM_HOME>\conf \watsonAssistant_naviga-tion.xml` directory under your Uni-ca Platform installation • Purpose: Import the `Affini-um|suite|uiNavigation|main-`	configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f "<PLATFORM_HOME>\conf\watsonAs-sistant_navigation.xml"

File details	Example command
`Menu|Analytics|`Watson Assistantconfiguration node configuration node	
• File: APISecurity.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the `Unica Platform	Security	API Management	Unica Platform	License node`	`configTool -vp -p "Affinium|suite|security" -f "<PLATFORM_HOME>\conf\upgrade12to121\APISecurity.xml"`
• APISecurity_Deliver.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the `Unica Platform	Security	API Management	Unica Deliver configuration node`	`configTool.bat -v -i -p "Affinium|suite|security|apiSecurity" -f "<PLATFORM_HOME>\conf\upgrade12to121\APISecurity_Deliver.xml"`
• LicenseDetailsNavigation.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the License details navigation option under the Settings menu	`configTool.bat -vp -p "Affinium| suite|uiNavigation|settingsMenu" -f "<PLATFORM_HOME>\conf\upgrade12to121\LicenseDeatilsNavigation.xml"`
File: APISecurity_GetInstalledApps.xml Location: `<PLATFORM_HOME>\conf\upgrade121to12103` Purpose: Import new API configuration	`configTool.bat -vp -p "Affinium|suite|security|apiSecurity|manager" -f APISecurity_GetInstalledApps.xml`

File details	Example command
File: PlanJourneyIntegrationOption.xml Location: `<PLATFORM_HOME>\conf\upgrade121to12103` Purpose: Import Plan Journey integration property	`configTool.bat -vp -p "Affinium|suite" -f PlanJourneyIntegrationOption.xml`
File: `SiteMinder_AdditionalHeaderVariables.xml` Location: `<PLATFORM_HOME>\conf\upgrade121to12103` Purpose: Import new property to accept additional header variables for Siteminder	`configTool.bat -vp -p "Affinium|suite|security|loginModes|siteMinderPartitionLogin" -f SiteMinder_AdditionalHeaderVariables.xml`
File: `PasswordRules.xml` Location: `<PLATFORM_HOME>\conf\upgrade12103to12104` Purpose: Import new password rules	`configTool.bat -vp -p "Affinium|Manager" -f PasswordRules.xml`
File: `support_Collaborate.xml` Location: `<PLATFORM_HOME>\conf\upgrade12103to12104` Purpose: Update CollaborateSupportServer property with new value	`configTool.bat -i -o -p "Affinium|suite" -f support_Collaborate.xml`
File: `LookerIntegration.xml` Location: `<PLATFORM_HOME>\conf\upgrade12104to1211` Purpose: Import Google Looker integration configuration	`configtool.bat -vp -p "Affinium|Report|integrations" -f LookerIntegration.xml`

File details	Example command

File: `OpenInsights_navigation.xml`
 	
Location: `<PLATFORM_HOME>\conf`
`\upgrade12104to1211`
 	
Purpose: Import navigation configuration for Open Insights menu
	`configtool.bat -vp -p "Affini-`
	`um|suite|uiNavigation|mainMenu|An-`
	`alytics" -f OpenInsights_naviga-`
	`tion.xml`

File: `OpenInsightsSeparator_naviga-`
`tion.xml`
 	
Location: `<PLATFORM_HOME>\conf`
`\upgrade12104to1211`
 	
Purpose: Import configuration for adding separator for Open Insights menu
	`configtool.bat -vp -p "Affini-`
	`um|suite|uiNavigation|mainMenu|An-`
	`alytics" -f OpenInsightsSeparator_-`
	`navigation.xml`

9. After you have imported the configuration properties in the previous step, restart the web application server on which Unica Platform is deployed so the changes are applied.

10. Complete the following steps to update the **Help > About** page.

    a. Use the **configTool** utility to export the **Affinium | Manager | about** category.

    > **Note:** The **Affinium | Manager | about** category is not visible on the **Configuration** page because it is marked hidden.

    Example (Windows™):

    ```
    configTool.bat -x -p "Affinium|Manager|about" -f "C:\Unica\Platform
    \conf\about.xml"
    ```

    b. Edit the exported XML file that you just created (`about.xml` in the example) to change the version number and display name.

Find the `releaseNumber` property and change the value to the current version of Unica Platform. Change the value of the `copyright` property to the appropriate copyright year.

In the following example, change the release number from 9.0.0.0.0 to 12.1.2.0.0 and change the copyright to 2021.

```
<property name="releaseNumber" type="string">

<displayNameKey>about.releaseNumber</displayNameKey>

<value>9.0.0.0.0.build_number</value>

</property>

</property>

<property id="541" name="copyright" type="string_property"

 width="40">

<value>2016</value>

</property>
```

c. Use the **configTool** utility to import the revised file.

You must use the `-o` option to overwrite the node. You must specify the parent node when you import.

Example (Windows™):

```
configTool.bat -vp -i -p "Affinium|Manager" -f "C:\Unica\Platform
\conf\about.xml" -o
```

11. Use the quartzjobtool to update scheduler jobs. This is a required step. If this upgrade tool is not run, any existing scheduled job fails 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

12. Deploy and verify your installation as described in the Deploying Unica Platform *(on page 168)*.

Chapter 6. Upgrading from version 9.1.0 with manual migration

The Unica Platform upgrade installer can perform all of the data migration that is required for an upgrade automatically. However, if the policy of your organization does not allow automatic migration, you must complete the migration procedure to upgrade Unica Platform manually.

Ensure that you have the following installers in the same directory.

- The master installer
- The Unica Platform installer

To upgrade to Unica Platform version 12.1.2 manually, you must run SQL scripts and run some command line utilities to populate the system tables. Ensure that your installation of Unica Platform is fully functional so that you can run the utilities. Complete information about using these utilities, including example commands for common tasks, is available in the following topics.

- populateDb *(on page 198)*
- configTool *(on page 184)*

The utilities are in the `tools\bin` directory under your Unica Platform installation.

See Unica Platform upgrade scenarios (on page 23) for information about upgrading from other versions.

1. Create a back-up copy of your Unica Platform system table database.

   ⚠️ **Important:** Do not skip this step. If the upgrade operation fails, you cannot roll back your database and your data is corrupted.

2. Undeploy your Unica Platform deployment.
3. Run the Unica master installer.

The Unica master installer starts. Use the following instructions after theUnica master installer starts.

- When the Unica master installer prompts you to choose an installation directory, choose the root installation directory, not the Unica Platform installation directory that is under this root directory.
- When the Unica master installer prompts you to enter Unica Platform database connection information, enter the information that pertains to your current Unica Platform system tables.

The Unica master installer pauses and starts the Unica Platform installer.

4. Complete the following steps while the Unica Platform installer is running.

   a. When the Unica Platform installer prompts you for an installation directory, select the directory of your current Unica Platform installation, usually named `Platform`.

   b. Allow the installer to create a back-up copy of your previous installation of Unica Platform.

   c. If Contact Central is selected in component selection during installation, select database type.

      > 📝 **Note:** Contact Central is supported with OneDb and Oracle database.

   d. Provide Contact Central database connection details. Notes. By default only automatic database setup will be executed for Control Center.

   e. Select **Manual database setup**.

   f. Clear the **Run Platform configuration** check box.

   g. Follow the remaining steps in the installation wizard and enter the requested information.

5. Run the following scripts against your system tables, where *DB_Type* is the database type.

Table 12. SQL scripts for upgrades from version 9.1.0

File details	File location
`ManagerSchema_DB_Type_911upg.sql`	`db\upgrade91to911` directory under your Unica Platform installation
`ManagerSchema_DB_Type_10upg.sql`	`db\upgrade912to10` directory under your Unica Platform installation
`DB_Type_QRTZ_Scheduler_10_upgrade_-Script.sql`	`db\upgrade912to10` directory under your Unica Platform installation
`ManagerSchema_DB_Type_10002upg.sql`	`db\upgrade10001to10002` directory under your Unica Platform installation
`ManagerSchema_DB_Type_101upg.sql`	`db\upgrade10002to101` directory under your Unica Platform installation
`ManagerSchema_DB_Type_11upg.sql`	`db\upgrade101to11` directory under your Unica Platform installation
`ManagerSchema_DB_Type_111upg.sql`	`db\upgrade1101to111` directory under your Unica Platform installation
`ManagerSchema_DB_Type_11102upg.sql`	`db\upgrade1111to11102` directory under your Unica Platform installation
`ManagerSchema_DB_Type_12upg.sql`	`db\upgrade1102to12` directory under your Unica Platform installation
`ManagerSchema_DB_Type_121upg.sql`	`db\upgrade12to121` directory under your Unica Platform installation
`ManagerUpdate_12101upg.sql`	`db\upgrade121to12101` directory under your Unica Platform installation
`ManagerUpdate_12103_DB_Type_upg.sql`	`db\upgrade12101to12103` directory under your Unica Platform installation

File details	File location
`ManagerUpdate_12104_DB_Type_upg.sql` and `DB_Type_QRTZ_Scheduler_12104_upgrade_Script.sql`	`db\upgrade12103to12104` directory under your Unica Platform installation
`ManagerUpdate_1211_DB_Type_upg.sql`	`db\upgrade12104to1211` directory under your Unica Platform installation
`ManagerUpdate_1212_DB_Type_upg.sql` and `ContentIntegrationUpdate_1212_DB_Type_upg.sql`	`db\upgrade1211to1212`directory under your Unica Platform installation

6. Use the `populateDb` utility to populate the system tables with default Unica Platform configuration properties, users and groups, and security roles and permissions.

   The `populateDb` utility is in the `tools\bin` directory under your Unica Platform installation. Example command: `populateDb -n Manager`

7. Use the configTool utility to import configuration properties as described below.

> ⚠️ **Important:** Perform the imports in the order shown in the following table.

Table 13. Configuration properties for upgrades from version 9.1.0

File details	Example command
• File: `AuditEvents.xml` • Location: `conf\upgrade91to911` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Audit Events** configuration node	configTool.bat -i -p "Affinium|suite" -f C:\Unica\Platform\conf\upgrade91to911\AuditEvents.xml

File details	Example command
• File: `FederatedAuthentica-tion.xml` • Location: `conf\up-grade91to911` directory under your Unica Platform installation • Purpose: Import the **Unica Plat-form | Security | Federated Au-thentication** configuration node	configTool.bat -i -p "Affinium|suite|se-curity" -f C:\Unica\Platform\conf\up-grade91to911\ FederatedAuthentica-tion.xml
• File: `MO_bulk_deactivation_-scheduler.xml` • Location: `conf\upgrade91to911` directory under your Unica Plat-form installation • Purpose: Import the configura-tion node for the bulk deactivation schedule feature for Unica Plan	configTool.bat -i -p "Affinium|suite|sched-uler| taskRegistrations" -f C:\Unica\Plat-form\conf\upgrade91to911\MO_bulk_de-activation_scheduler.xml
• File: `emm_audit_navigation.xml` • Location: `conf` directory under your Unica Platform installation • Purpose: Import the **Analytics > Platform** menu item for Audit Event reports	configTool.bat -i -p "Affinium|suite|uiNav-igation|mainMenu|Analytics" -f C:\Uni-ca\Platform\conf\emm_audit_naviga-tion.xml
• File: `APISecurity.xml` • Location: `conf\upgrade911to912` directory under your Unica Plat-form installation • Purpose: Import the **Unica Plat-form | Security | API management** configuration node	configTool.bat -i -p "Affinium|suite|se-curity" -f C:\Unica\Platform\conf\up-grade911to912\APISecurity.xml

File details	Example command
• File: `APISecurity.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API management** configuration node	configTool.bat -vp -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade912to10\APISecurity.xml Note that this version of the `APISecurity.xml` file is different from the one shown in the previous row and is in a different location.
• File: `SAMLAuthentication.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | SAML2.0** configuration node	configTool.bat -vp -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade912to10\SAMLAuthentication.xml
• File: `SAMLAuthenticationDetails.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | Login method details | SAML 2.0** configuration nodes	configTool.bat -vp -p "Affinium|suite|security|loginModes" -f C:\Unica\Platform\conf\upgrade912to10\SAMLAuthenticationDetails.xml
• File: `ExternalScheduler.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation	configTool.bat -vp -p "Affinium|suite|scheduler| taskRegistrations" -f C:\Unica\Platform\conf\upgrade912to10\ExternalScheduler.xml

File details	Example command
• Purpose: Import the **Unica Platform | Scheduler | Scheduler registrations | Unica Platform** configuration node	
• File: `JWTAuthentication.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | JWT authentication** configuration node	configTool.bat -vp -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade912to10\JWTAuthentication.xml
• File: `SecureSuiteStaticContent.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Apply security on static content for all applications** property under the **Unica Platform** configuration node	configTool.bat -vp -p "Affinium|suite" -f C:\Unica\Platform\conf\upgrade912to10\SecureSuiteStaticContent.xml
• File: `APISecurity_interactCollection.xml` • Location: `conf\upgrade10to1001` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API Management | Unica Campaign** configuration node	configTool.bat -vp -p "Affinium|suite|security|apiSecurity|campaign" -f <Platform_Home>\conf\upgrade10to1001\APISecurity_interactCollection.xml

File details	Example command
• File: `APISecurity_triggeredMessages.xml` • Location: `conf\upgrade10to1001` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API Management | Unica Campaign** configuration node	configTool.bat -vp -p "Affinium|suite|security|apiSecurity|campaign" -f <Platform_Home>\conf\upgrade10to1001\APISecurity_triggeredMessages.xml
• File: `supportServer_config.xml` • Location: `conf\upgrade10to101` directory under your Unica Platform installation • Purpose: Import the **HCL Unica** configuration nodes	configTool.bat -vp -p "Affinium" -f <Platform_Home>\conf\upgrade10to101\supportServer_config.xml
• File: `cognos11_integration.xml` • Location: `conf\upgrade101to11` directory under your Unica Platform installation • Purpose: Import the **Affinium|Report|integrations|cognos11** configuration node	configTool.bat -vp -p "Affinium|Report|integrations" -f <Platform_Home>/conf/upgrade101to11/cognos11_integration.xml
• File: watsonAssistant_navigation.xml • Location: `<PLATFORM_HOME>\conf\watsonAssistant_navigation.xml` directory under your Unica Platform installation • Purpose: Import the `Affinium|suite|uiNavigation|main-`	configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f "<PLATFORM_HOME>\conf\watsonAssistant_navigation.xml"

File details	Example command
`Menu|Analytics|`Watson Assistantconfiguration node configuration node	
• File: APISecurity.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the `Unica Platform | Security | API Management | Unica Platform | License node`	`configTool -vp -p "Affinium|suite|security" -f "<PLATFORM_HOME>\conf\upgrade12to121\APISecurity.xml"`
• APISecurity_Deliver.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the `Unica Platform | Security | API Management | Unica Deliver configuration node`	`configTool.bat -v -i -p "Affinium|suite|security|apiSecurity" -f "<PLATFORM_HOME>\conf\upgrade12to121\APISecurity_Deliver.xml"`
• LicenseDetailsNavigation.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the License details navigation option under the Settings menu	`configTool.bat -vp -p "Affinium| suite|uiNavigation|settingsMenu" -f "<PLATFORM_HOME>\conf\upgrade12to121\LicenseDeatilsNavigation.xml"`

File details	Example command
• File: APISecurity_GetInstalledApps.xml • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import new API configuration	`configTool.bat -vp -p "Affinium|suite|security|apiSecurity|manager" -f APISecurity_GetInstalledApps.xml`
• File: PlanJourneyIntegrationOption.xml • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import Plan Journey integration property	`configTool.bat -vp -p "Affinium|suite" -f PlanJourneyIntegrationOption.xml`
• File: `SiteMinder_AdditionalHeaderVariables.xml` • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import new property to accept additional header variables for Siteminder	`configTool.bat -vp -p "Affinium|suite|security|loginModes|siteMinderPartitionLogin" -f SiteMinder_AdditionalHeaderVariables.xml`
• File: `PasswordRules.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12103to12104` • Purpose: Import new password rules	`configTool.bat -vp -p "Affinium|Manager" -f PasswordRules.xml`

File details	Example command
• File: `support_Collaborate.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12103to12104` • Purpose: Update CollaborateSupportServer property with new value	`configTool.bat -i -o -p "Affinium|suite" -f support_Collaborate.xml`
• File: `LookerIntegration.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12104to1211` • Purpose: Import Google Looker integration configuration	`configtool.bat -vp -p "Affinium|Report|integrations" -f LookerIntegration.xml`
• File: `OpenInsights_navigation.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12104to1211` • Purpose: Import navigation configuration for Open Insights menu	`configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f OpenInsights_navigation.xml`
• File: `OpenInsightsSeparator_navigation.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12104to1211` • Purpose: Import configuration for adding separator for Open Insights menu	`configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f OpenInsightsSeparator_navigation.xml`

8. After you have imported the configuration properties in the previous step, restart the web application server on which Unica Platform is deployed so the changes are applied.

9. Complete the following steps to update the **Help > About** page.

   a. Use the **configTool** utility to export the **Affinium | Manager | about** category.

      > 📝 **Note:** The **Affinium | Manager | about** category is not visible on the **Configuration** page because it is marked hidden.

      Example (Windows™):

      ```
      configTool.bat -x -p "Affinium|Manager|about" -f "C:\Unica\Platform
      \conf\about.xml"
      ```

   b. Edit the exported XML file that you just created (`about.xml` in the example) to change the version number and display name.

      Find the `releaseNumber` property and change the value to the current version of Unica Platform. Change the value of the `copyright` property to the appropriate copyright year.

      In the following example, change the release number from 9.1.0.0.0 to 12.1.2.0.0 and change the copyright to 2021.

      ```
      <property name="releaseNumber" type="string">

      <displayNameKey>about.releaseNumber</displayNameKey>

      <value>9.1.0.0.0.build_number</value>

      </property>

      </property>

      <property id="541" name="copyright" type="string_property"
       width="40">

      <value>2016</value>

      </property>
      ```

   c. Use the **configTool** utility to import the revised file.

You must use the `-o` option to overwrite the node. You must specify the parent node when you import.

Example (Windows™):

```
configTool.bat -vp -i -p "Affinium|Manager" -f "C:\Unica\Platform
\conf\about.xml" -o
```

10. Use the quartzjobtool to update scheduler jobs. This is a required step. If this upgrade tool is not run, any existing scheduled job fails 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

11. Deploy and verify your installation as described in the Deploying Unica Platform *(on page 168)*.

Chapter 7. Upgrading from version 9.1.1 with manual migration

The Unica Platform upgrade installer can perform all of the data migration that is required for an upgrade automatically. However, if the policy of your organization does not allow automatic migration, you must complete the migration procedure to upgrade Unica Platform manually.

Ensure that you have the following installers in the same directory.

- The master installer
- The Unica Platform installer

To upgrade to Unica Platform version 12.1.2 manually, you must run SQL scripts and run some command line utilities to populate the system tables. Ensure that your installation of Unica Platform is fully functional so that you can run the utilities. Complete information about using these utilities, including example commands for common tasks, is available in the following topics.

- populateDb *(on page 198)*
- configTool *(on page 184)*

The utilities are in the `tools\bin` directory under your Unica Platform installation.

See Unica Platform upgrade scenarios (on page 23) for information about upgrading from other versions.

1. Create a back-up copy of your Unica Platform system table database.

    ⚠️ **Important:** Do not skip this step. If the upgrade operation fails, you cannot roll back your database and your data is corrupted.

2. Undeploy your Unica Platform deployment.
3. Run the Unica master installer.

The Unica master installer starts. Use the following instructions after theUnica master installer starts.

- When the Unica master installer prompts you to choose an installation directory, choose the root installation directory, not the Unica Platform installation directory that is under this root directory.
- When the Unica master installer prompts you to enter Unica Platform database connection information, enter the information that pertains to your current Unica Platform system tables.

The Unica master installer pauses and starts the Unica Platform installer.

4. Complete the following steps while the Unica Platform installer is running.

   a. When the Unica Platform installer prompts you for an installation directory, select the directory of your current Unica Platform installation, usually named `Platform`.

   b. Allow the installer to create a back-up copy of your previous installation of Unica Platform.

   c. If Contact Central is selected in component selection during installation, select database type.

      > ✏️ **Note:** Contact Central is supported with OneDb and Oracle database.

   d. Provide Contact Central database connection details. Notes. By default only automatic database setup will be executed for Control Center.

   e. Select **Manual database setup**.

   f. Clear the **Run Platform configuration** check box.

   g. Follow the remaining steps in the installation wizard and enter the requested information.

5. Run the following scripts against your system tables, where *DB_Type* is the database type.

Table 14. SQL scripts for upgrades from version 9.1.1

File details	File location
`ManagerSchema_DB_Type_10upg.sql`	`\db\upgrade912to10` directory under your Unica Platform installation
`DB_Type_QRTZ_Scheduler_10_upgrade_-Script.sql`	`db\upgrade912to10` directory under your Unica Platform installation
`ManagerSchema_DB_Type_10002upg.sql`	`db\upgrade10001to10002` directory under your Unica Platform installation
`ManagerSchema_DB_Type_101upg.sql`	`db\upgrade10002to101` directory under your Unica Platform installation
`ManagerSchema_DB_Type_11upg.sql`	`db\upgrade101to11` directory under your Unica Platform installation
`ManagerSchema_DB_Type_111upg.sql`	`db\upgrade1101to111` directory under your Unica Platform installation
`ManagerSchema_DB_Type_11102upg.sql`	`db\upgrade1111to11102 directory` under your Unica Platform installation
`ManagerSchema_DB_Type_12upg.sql`	`db\upgrade1102to12 directory` under your Unica Platform installation
`ManagerSchema_DB_Type_121upg.sql`	`db\upgrade12to121` directory under your Unica Platform installation
`ManagerUpdate_12101upg.sql`	`db\upgrade121to12101` directory under your Unica Platform installation
`ManagerUpdate_12103_DB_Type_upg.sql`	`db\upgrade12101to12103` directory under your Unica Platform installation
`ManagerUpdate_12104_DB_Type_upg.sql` and `DB_Type_QRTZ_Scheduler_12104_-upgrade_Script.sql`	`db\upgrade12103to12104` directory under your Unica Platform installation

File details	File location
`ManagerUpdate_1211_DB_Type_upg.sql`	`db\upgrade12104to1211` directory under your Unica Platform installation
`ManagerUpdate_1212_DB_Type_upg.sql` and `ContentIntegrationUpdate_1212_-DB_Type_upg.sql`	`db\upgrade1211to1212` directory under your Unica Platform installation

6. Use the `populateDb` utility to populate the system tables with default Unica Platform configuration properties, users and groups, and security roles and permissions.

   The `populateDb` utility is in the `tools\bin` directory under your Unica Platform installation. Example command: `populateDb -n Manager`

7. Use the configTool utility to import configuration properties as described below.

   > **Important:** Perform the imports in the order shown in the following table.

Table 15. Configuration properties for upgrades from version 9.1.1

File details	Example command
• File: `emm_audit_navigation.xml` • Location: `conf` directory under your Unica Platform installation • Purpose: Import the **Analytics > Platform** menu item for Audit Event reports	configTool.bat -i -p "Affinium|suite|uiNavigation|mainMenu| Analytics" -f C:\Unica\Platform\conf\ emm_audit_navigation.xml
• File: `APISecurity.xml` • Location: `conf\upgrade911to912` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API management** configuration node	configTool.bat -i -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade911to912\APISecurity.xml

File details	Example command
• File: `APISecurity.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API management** configuration node	configTool.bat -vp -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade912to10\APISecurity.xml Note that this version of the `APISecurity.xml` file is different from the one shown in the previous row and is in a different location
• File: `SAMLAuthentication.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | SAML2.0** configuration node	configTool.bat -vp -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade912to10\SAMLAuthentication.xml
• File: `SAMLAuthenticationDetails.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | Login method details | SAML 2.0** configuration nodes	configTool.bat -vp -p "Affinium|suite|security|loginModes" -f C:\Unica\Platform\conf\upgrade912to10\SAMLAuthenticationDetails.xml
• File: `ExternalScheduler.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation	configTool.bat -vp -p "Affinium|suite|scheduler| taskRegistrations" -f C:\Unica\Platform\conf\upgrade912to10\ExternalScheduler.xml

File details	Example command
• Purpose: Import the **Unica Platform | Scheduler | Scheduler registrations | Unica Platform** configuration node	
• File: `JWTAuthentication.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | JWT authentication** configuration node	configTool.bat -vp -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade912to10\JWTAuthentication.xml
• File: `SecureSuiteStaticContent.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Apply security on static content for all applications** property under the **Unica Platform** configuration node	configTool.bat -vp -p "Affinium|suite" -f C:\Unica\Platform\conf\upgrade912to10\SecureSuiteStaticContent.xml
• File: `APISecurity_interactCollection.xml` • Location: `conf\upgrade10to1001` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API Management | Unica Campaign** configuration node	configTool.bat -vp -p "Affinium|suite|security|apiSecurity|campaign" -f <Platform_Home>\conf\upgrade10to1001\APISecurity_interactCollection.xml

File details	Example command
• File: `APISecurity_triggeredMes-sages.xml` • Location: `conf\upgrade10to1001` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API Management | Unica Campaign** configuration node	configTool.bat -vp -p "Affinium|suite|security|apiSecurity|campaign" -f <Platform_Home>\conf\upgrade10to1001\APISecurity_triggered-Messages.xml
• File: `supportServer_config.xml` • Location: `conf\upgrade10to101` directory under your Unica Platform installation • Purpose: Import the **HCL Unica** configuration nodes	configTool.bat -vp -p "Affinium" -f <Platform_Home>\conf\upgrade10to101\supportServer_config.xml
• File: `cognos11_integration.xml` • Location: `conf\upgrade101to11` directory under your Unica Platform installation • Purpose: Import the **Affinium|Report|integrations|cognos11** configuration node	configTool.bat -vp -p "Affinium|Report|integrations" -f <Platform_Home>/conf/upgrade101to11/cognos11_integration.xml
• File: `watsonAssistant_navigation.xml` • Location: `<PLATFORM_-HOME>\conf\watsonAssistant_navigation.xml` directory under your Unica Platform installation • Purpose: Import the `Affinium|suite|uiNavigation|main-`	configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f "<PLATFORM_HOME>\conf\watsonAssistant_navigation.xml"

File details	Example command
`Menu|Analytics|`Watson Assistantconfiguration node configuration node	
• File: APISecurity.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the `Unica Platform | Security | API Management | Unica Platform | License node`	`configTool -vp -p "Affinium|suite|security" -f "<PLATFORM_HOME>\conf\upgrade12to121\APISecurity.xml"`
• APISecurity_Deliver.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the `Unica Platform | Security | API Management | Unica Deliver configuration node`	`configTool.bat -v -i -p "Affinium|suite|security|apiSecurity" -f "<PLATFORM_HOME>\conf\upgrade12to121\APISecurity_Deliver.xml"`
• LicenseDetailsNavigation.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the License details navigation option under the Settings menu	`configTool.bat -vp -p "Affinium| suite|uiNavigation|settingsMenu" -f "<PLATFORM_HOME>\conf\upgrade12to121\LicenseDeatilsNavigation.xml"`

File details	Example command
• File: APISecurity_GetInstalledApps.xml • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import new API configuration	`configTool.bat -vp -p "Affinium|suite|security|apiSecurity|manager" -f APISecurity_GetInstalledApps.xml`
• File: PlanJourneyIntegrationOption.xml • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import Plan Journey integration property	`configTool.bat -vp -p "Affinium|suite" -f PlanJourneyIntegrationOption.xml`
• File: `SiteMinder_AdditionalHeaderVariables.xml` • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import new property to accept additional header variables for Siteminder	`configTool.bat -vp -p "Affinium|suite|security|loginModes|siteMinderPartitionLogin" -f SiteMinder_AdditionalHeaderVariables.xml`
• File: `PasswordRules.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12103to12104` • Purpose: Import new password rules	`configTool.bat -vp -p "Affinium|Manager" -f PasswordRules.xml`

File details	Example command
• File: `support_Collaborate.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12103to12104` • Purpose: Update CollaborateSupportServer property with new value	`configTool.bat -i -o -p "Affinium|suite" -f support_Collaborate.xml`
• File: `LookerIntegration.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12104to1211` • Purpose: Import Google Looker integration configuration	`configtool.bat -vp -p "Affinium|Report|integrations" -f LookerIntegration.xml`
• File: `OpenInsights_navigation.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12104to1211` • Purpose: Import navigation configuration for Open Insights menu	`configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f OpenInsights_navigation.xml`
• File: `OpenInsightsSeparator_navigation.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12104to1211` • Purpose: Import configuration for adding separator for Open Insights menu	`configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f OpenInsightsSeparator_navigation.xml`

8. After you have imported the configuration properties in the previous step, restart the web application server on which Unica Platform is deployed so the changes are applied.

9. Complete the following steps to update the **Help > About** page.

   a. Use the **configTool** utility to export the **Affinium | Manager | about** category.

      > 📝 **Note:** The **Affinium | Manager | about** category is not visible on the **Configuration** page because it is marked hidden.

      Example (Windows™):

      ```
      configTool.bat -x -p "Affinium|Manager|about" -f "C:\Unica\Platform
      \conf\about.xml"
      ```

   b. Edit the exported XML file that you just created (`about.xml` in the example) to change the version number and display name.

      Find the `releaseNumber` property and change the value to the current version of Unica Platform. Change the value of the `copyright` property to the appropriate copyright year.

      In the following example, change the release number from 9.1.1.0.0 to 12.1.2.0.0 and change the copyright to 2021.

      ```
      <property name="releaseNumber" type="string">

      <displayNameKey>about.releaseNumber</displayNameKey>

      <value>9.1.1.0.0.build_number</value>

      </property>

      </property>

      <property id="541" name="copyright" type="string_property"
       width="40">

      <value>2016</value>

      </property>
      ```

   c. Use the **configTool** utility to import the revised file.

You must use the `-o` option to overwrite the node. You must specify the parent node when you import.

Example (Windows™):

```
configTool.bat -vp -i -p "Affinium|Manager" -f "C:\Unica\Platform
\conf\about.xml" -o
```

10. Add the quartjobtool.

11. Deploy and verify your installation as described in the Deploying Unica Platform *(on page 168)*.

Chapter 8. Upgrading from version 9.1.2 with manual migration

The Unica Platform upgrade installer can perform all of the data migration that is required for an upgrade automatically. However, if the policy of your organization does not allow automatic migration, you must complete the migration procedure to upgrade Unica Platform manually.

Ensure that you have the following installers in the same directory.

- The Unica master installer
- The Unica Platform installer

To upgrade to Unica Platform version 12.1.2 manually, you must run SQL scripts and run some command line utilities to populate the system tables. Ensure that your installation of Unica Platform is fully functional so that you can run the utilities. Complete information about using these utilities, including example commands for common tasks, is available in the following topics.

- populateDb *(on page 198)*
- configTool *(on page 184)*

The utilities are in the `tools\bin` directory under your Unica Platform installation.

See Unica Platform upgrade scenarios (on page 23) for information about upgrading from other versions.

1. Create a back-up copy of your Unica Platform system table database.

   ⚠️ **Important:** Do not skip this step. If the upgrade operation fails, you cannot roll back your database and your data is corrupted.

2. Undeploy your Unica Platform deployment.
3. Run the Unica master installer.

The Unica master installer starts. Use the following instructions after theUnica master installer starts.

- When the Unica master installer prompts you to choose an installation directory, choose the root installation directory, not the Unica Platform installation directory that is under this root directory.
- When the Unica master installer prompts you to enter Unica Platform database connection information, enter the information that pertains to your current Unica Platform system tables.

The Unica master installer pauses and starts the Unica Platform installer.

4. Complete the following steps while the Unica Platform installer is running.

    a. When the Unica Platform installer prompts you for an installation directory, select the directory of your current Unica Platform installation, usually named `Platform`.

    b. Allow the installer to create a back-up copy of your previous installation of Unica Platform.

    c. If Contact Central is selected in component selection during installation, select database type.

    > ✏️ **Note:** Contact Central is supported with OneDb and Oracle database.

    d. Provide Contact Central database connection details. Notes. By default only automatic database setup will be executed for Control Center.

    e. Select **Manual database setup**.

    f. Clear the **Run Platform configuration** check box.

    g. Follow the remaining steps in the installation wizard and enter the requested information.

5. Run the following scripts against your system tables, where *DB_Type* is the database type.

Table 16. SQL scripts for upgrades from version 9.1.2

File details	File location
`ManagerSchema_DB_Type_10upg.sql`	`db\upgrade912to10` directory under your Unica Platform installation
`DB_Type_QRTZ_Scheduler_10_upgrade_-Script.sql`	`db\upgrade912to10` directory under your Unica Platform installation
`ManagerSchema_DB_Type_10002upg.sql`	`db\upgrade10001to10002` directory under your Unica Platform installation
`ManagerSchema_DB_Type_101upg.sql`	`db\upgrade10002to101` directory under your Unica Platform installation
`ManagerSchema_DB_Type_11upg.sql`	`db\upgrade101to11` directory under your Unica Platform installation
`ManagerSchema_DB_Type_111upg.sql`	`db\upgrade1101to111` directory under your Unica Platform installation
`ManagerSchema_DB_Type_11102upg.sql`	`db\upgrade1111to11102 directory` under your Unica Platform installation
`ManagerSchema_DB_Type_12upg.sql`	`db\upgrade1102to12 directory` under your Unica Platform installation
`ManagerSchema_DB_Type_121upg.sql`	`db\upgrade12to121` directory under your Unica Platform installation
`ManagerUpdate_12101upg.sql`	`db\upgrade121to12101` directory under your Unica Platform installation
`ManagerUpdate_12103_DB_Type_upg.sql`	`db\upgrade12101to12103` directory under your Unica Platform installation
`ManagerUpdate_12104_DB_Type_upg.sql` and `DB_Type_QRTZ_Scheduler_12104_-upgrade_Script.sql`	`db\upgrade12103to12104` directory under your Unica Platform installation

File details	File location
`ManagerUpdate_1211_DB_Type_upg.sql`	`db\upgrade12104to1211` directory under your Unica Platform installation
`ManagerUpdate_1212_DB_Type_upg.sql` and `ContentIntegrationUpdate_1212_-DB_Type_upg.sql`	`db\upgrade1211to1212`directory under your Unica Platform installation

6. Use the `populateDb` utility to populate the system tables with default Unica Platform configuration properties, users and groups, and security roles and permissions.

   The `populateDb` utility is in the `tools\bin` directory under your Unica Platform installation. Example command: `populateDb -n Manager`

7. Use the configTool utility to import configuration properties as described below.

   ⚠️ **Important:** Perform the imports in the order shown in the following table.

Table 17. Configuration properties for upgrades from version 9.1.2

File details	Example command
• File: `APISecurity.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API management** configuration node	configTool.bat -vp -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade912to10\APISecurity.xml
• File: `SAMLAuthentication.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | SAML2.0** configuration node	configTool.bat -vp -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade912to10\SAMLAuthentication.xml

File details	Example command
• File: `SAMLAuthenticationDetails.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | Login method details | SAML 2.0** configuration nodes	configTool.bat -vp -p "Affinium|suite|security|loginModes" -f C:\Unica\Platform\conf\upgrade912to10\SAMLAuthenticationDetails.xml
• File: `ExternalScheduler.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Scheduler | Scheduler registrations | Unica Platform** configuration node	configTool.bat -vp -p "Affinium|suite|scheduler| taskRegistrations" -f C:\Unica\Platform\conf\upgrade912to10\ExternalScheduler.xml
• File: `JWTAuthentication.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | JWT authentication** configuration node	configTool.bat -vp -p "Affinium|suite|security" -f C:\Unica\Platform\conf\upgrade912to10\JWTAuthentication.xml
• File: `SecureSuiteStaticContent.xml` • Location: `conf\upgrade912to10` directory under your Unica Platform installation	configTool.bat -vp -p "Affinium|suite" -f C:\Unica\Platform\conf\upgrade912to10\SecureSuiteStaticContent.xml

File details	Example command
• Purpose: Import the **Apply security on static content for all applications** property under the **Unica Platform** configuration node	
• File: `APISecurity_interactCollection.xml` • Location: `conf\upgrade10to1001` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API Management | Unica Campaign** configuration node	configTool.bat -vp -p "Affinium|suite|security|apiSecurity|campaign" -f <Platform_Home>\conf\upgrade10to1001\APISecurity_interactCollection.xml
• File: `APISecurity_triggeredMessages.xml` • Location: `conf\upgrade10to1001` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API Management | Unica Campaign** configuration node	```configTool.bat -vp -p "Affinium|suite|security|apiSecurity|campaign" -f <Platform_Home>\conf\upgrade10to1001\APISecurity_triggeredMessages.xml```
• File: `supportServer_config.xml` • Location: `conf\upgrade10to101` directory under your Unica Platform installation • Purpose: Import the **HCL Unica** configuration nodes	configTool.bat -vp -p "Affinium" -f <Platform_Home>\conf\upgrade10to101\supportServer_config.xml

File details	Example command
• File: `cognos11_integration.xml` • Location: `conf\upgrade101to11` directory under your Unica Platform installation • Purpose: Import the **Affinium|Report|integrations|cognos11** configuration node	configTool.bat -vp -p "Affinium|Report|integrations" -f <Platform_Home>/conf/upgrade101to11/cognos11_integration.xml
• File: `watsonAssistant_navigation.xml` • Location: `<PLATFORM_HOME>\conf\watsonAssistant_navigation.xml` directory under your Unica Platform installation • Purpose: Import the `Affinium|suite|uiNavigation|mainMenu|Analytics|`Watson Assistantconfiguration node configuration node	configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f "<PLATFORM_HOME>\conf\watsonAssistant_navigation.xml"
• File: APISecurity.xml • Location: `<PLATFORM_HOME>\conf\upgrade12to121` • Purpose: Import the `Unica Platform | Security | API Management | Unica Platform | License node`	```configTool -vp -p "Affinium|suite|security" -f "<PLATFORM_HOME>\conf\upgrade12to121\APISecurity.xml"```
• APISecurity_Deliver.xml • Location: `<PLATFORM_HOME>\conf\upgrade12to121`	```configTool.bat -v -i -p "Affinium|suite|security|apiSecurity" -f "<PLATFORM_HOME>\conf\up-```

File details	**Example command**
• Purpose: Import the `Unica Plat-form	Security	API Manage-ment	Unica Deliver configu-ration node`	`grade12to121\APISecurity_Deliv-er.xml"`
• LicenseDetailsNavigation.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the License details navigation option under the Set-tings menu	`configTool.bat -vp -p "Affini-um| suite|uiNavigation|settings-Menu" -f "<PLATFORM_HOME>\conf\up-grade12to121\LicenseDeatilsNaviga-tion.xml"`
• File: APISecurity_GetInstalledApp-s.xml • Location: `<PLATFOR-M_HOME>\conf \up-grade121to12103` • Purpose: Import new API configura-tion	`configTool.bat -vp -p "Affini-um|suite|security|apiSecurity|man-ager" -f APISecurity_GetInstalled-Apps.xml`
• File: PlanJourneyIntegrationOp-tion.xml • Location: `<PLATFOR-M_HOME>\conf \up-grade121to12103` • Purpose: Import Plan Journey inte-gration property	`configTool.bat -vp -p "Affini-um|suite" -f PlanJourneyIntegra-tionOption.xml`

File details	Example command
• File: `SiteMinder_AdditionalHeaderVariables.xml` • Location: `<PLATFOR­M_HOME>\conf \upgrade121to12103` • Purpose: Import new property to accept additional header variables for Siteminder	`configTool.bat -vp -p "Affinium|suite|security|loginModes|siteMinderPartitionLogin" -f SiteMinder_AdditionalHeaderVariables.xml`
• File: `PasswordRules.xml` • Location: `<PLATFOR­M_HOME>\conf \upgrade12103to12104` • Purpose: Import new password rules	`configTool.bat -vp -p "Affinium|Manager" -f PasswordRules.xml`
• File: `support_Collaborate.xml` • Location: `<PLATFOR­M_HOME>\conf \upgrade12103to12104` • Purpose: Update CollaborateSupportServer property with new value	`configTool.bat -i -o -p "Affinium|suite" -f support_Collaborate.xml`
• File: `LookerIntegration.xml` • Location: `<PLATFOR­M_HOME>\conf \upgrade12104to1211` • Purpose: Import Google Looker integration configuration	`configtool.bat -vp -p "Affinium|Report|integrations" -f LookerIntegration.xml`

File details	Example command
• File: `OpenInsights_naviga-tion.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade12104to1211` • Purpose: Import navigation config-uration for Open Insights menu	`configtool.bat -vp -p "Affini-um|suite|uiNavigation|mainMenu|An-alytics" -f OpenInsights_naviga-tion.xml`
• File: `OpenInsightsSeparator_nav-igation.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade12104to1211` • Purpose: Import configuration for adding separator for Open Insights menu	`configtool.bat -vp -p "Affini-um|suite|uiNavigation|mainMenu|An-alytics" -f OpenInsightsSeparator_-navigation.xml`

8. After you have imported the configuration properties in the previous step, restart the web application server on which Unica Platform is deployed so the changes are applied.

9. Complete the following steps to update the **Help > About** page.

    a. Use the **configTool** utility to export the **Affinium | Manager | about** category.

    > **Note:** The **Affinium | Manager | about** category is not visible on the **Configuration** page because it is marked hidden.

    Example (Windows™):

    ```
    configTool.bat -x -p "Affinium|Manager|about" -f "C:\Unica\Platform
    \conf\about.xml"
    ```

    b. Edit the exported XML file that you just created (`about.xml` in the example) to change the version number and display name.

Find the `releaseNumber` property and change the value to the current version of Unica Platform. Change the value of the `copyright` property to the appropriate copyright year.

In the following example, change the release number from 9.1.2.0.0 to 12.1.2.0.0 and change the copyright to 2021.

```
<property name="releaseNumber" type="string">

<displayNameKey>about.releaseNumber</displayNameKey>

<value>9.1.2.0.0.build_number</value>

</property>

</property>

<property id="541" name="copyright" type="string_property"

 width="40">

<value>2016</value>

</property>
```

c. Use the **configTool** utility to import the revised file.

You must use the `-o` option to overwrite the node. You must specify the parent node when you import.

Example (Windows™):

```
configTool.bat -vp -i -p "Affinium|Manager" -f "about.xml" -o
```

10. Use the quartzjobtool to update scheduler jobs. This is a required step. If this upgrade tool is not run, any existing scheduled job fails 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`

11. Deploy and verify your installation as described in the Deploying Unica Platform *(on page 168)*.

Chapter 9. Upgrading from version 10.0.0 with manual migration

The Unica Platform upgrade installer can perform all of the data migration that is required for an upgrade automatically. However, if the policy of your organization does not allow automatic migration, you must complete the migration procedure to upgrade Unica Platform manually.

Ensure that you have the following installers in the same directory.

- The Unica master installer
- The Unica Platform installer

To upgrade to Unica Platform version 12.1.2 manually, you must run SQL scripts and run some command line utilities to populate the system tables. Ensure that your installation of Unica Platform is fully functional so that you can run the utilities. Complete information about using these utilities, including example commands for common tasks, is available in the following topics.

- populateDb *(on page 198)*
- configTool *(on page 184)*

The utilities are in the `tools\bin` directory under your Unica Platform installation.

See Unica Platform upgrade scenarios (on page 23) for information about upgrading from other versions.

1. Create a back up copy of your Unica Platform system table database.

   ⚠️ **Important:** Do not skip this step. If the upgrade operation fails, you cannot roll back your database and your data is corrupted.

2. Undeploy your Unica Platform deployment.
3. Run the Unica master installer.

The Unica master installer starts. Use the following instructions after theUnica master installer starts.

- When the Unica master installer prompts you to choose an installation directory, choose the root installation directory, not the Unica Platform installation directory that is under this root directory.
- When the Unica master installer prompts you to enter Unica Platform database connection information, enter the information that pertains to your current Unica Platform system tables.

The Unica master installer pauses and starts the Unica Platform installer.

4. Complete the following steps while the Unica Platform installer is running.

   a. When the Unica Platform installer prompts you for an installation directory, select the directory of your current Unica Platform installation, usually named `Platform`.

   b. Allow the installer to create a back-up copy of your previous installation of Unica Platform.

   c. If Contact Central is selected in component selection during installation, select database type.

      > **Note:** Contact Central is supported with OneDb and Oracle database.

   d. Provide Contact Central database connection details. Notes. By default only automatic database setup will be executed for Control Center.

   e. Select **Manual database setup**.

   f. Clear the **Run Platform configuration** check box.

   g. Follow the remaining steps in the installation wizard and enter the requested information.

5. Execute quartzjobtool.sh / bat utility. This utility must be executed to update the Quartz Jobs (existing schedules) in the Unica application. Perfrom the following steps to execute the utility:

   a. Navigate to `PLATFORM_HOME/tools/bin`.

   b. Run `quartzjobtool.sh /bat`.

   This script execution enables the existing schedules to be compatible with v12.0
   Unica applications.

6. Deploy your installation.

7. Run the following scripts against your system tables, where *DB_Type* is the database
   type.

Table 18. SQL scripts for upgrades from version 10.0.0

File details	File location
`ManagerSchema_DB_Type_10002upg.sql`	`db\upgrade10001to10002` directory under your Unica Platform installation
`ManagerSchema_DB_Type_101upg.sql`	`db\upgrade10002to10`1 directory under your Unica Platform installation
`ManagerSchema_DB_Type_11upg.sql`	`db\upgrade101to11` directory under your Unica Platform installation
`ManagerSchema_DB_Type_111upg.sql`	`db\upgrade1101to111` directory under your Unica Platform installation
`ManagerSchema_DB_Type_11102upg.sql`	`db\upgrade1111to11102 directory` under your Unica Platform installation
`ManagerSchema_DB_Type_12upg.sql`	`db\upgrade1102to12 directory` under your Unica Platform installation
`ManagerSchema_DB_Type_121upg.sql`	`db\upgrade12to121` directory under your Unica Platform installation
`ManagerUpdate_12101upg.sql`	`db\upgrade121to12101` directory under your Unica Platform installation
`ManagerUpdate_12103_DB_Type_upg.sql`	`db\upgrade12101to12103` directory under your Unica Platform installation

File details	File location
`ManagerUpdate_12104_DB_Type_upg.sql` and `DB_Type_QRTZ_Scheduler_12104_-upgrade_Script.sql`	`db\upgrade12103to12104` directory under your Unica Platform installation
`ManagerUpdate_1211_DB_Type_upg.sql`	`db\upgrade12104to1211` directory under your Unica Platform installation
`ManagerUpdate_1212_DB_Type_upg.sql` and `ContentIntegrationUpdate_1212_-DB_Type_upg.sql`	`db\upgrade1211to1212`directory under your Unica Platform installation

8. Use the `populateDb` utility to populate the system tables with default Unica Platform configuration properties, users and groups, and security roles and permissions.

   The `populateDb` utility is in the `tools\bin` directory under your Unica Platform installation. Example command: `populateDb -n Manager`

9. Use the configTool utility to import configuration properties as described below.

   ❗ **Important:** Perform the imports in the order shown in the following table.

Table 19. Configuration properties for upgrades from version 10.0.0

File details	Example command
• File: `APISecurity_interactCol-lection.xml` • Location: `conf\upgrade10to1001` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API Management | Unica Campaign** configuration node	configTool.bat -vp -p "Affinium|suite|security|apiSecurity|campaign" -f <Platform_Home>\conf\up-grade10to1001\APISecurity_interactCol-lection.xml

File details	Example command
• File: `APISecurity_triggeredMes-sages.xml` • Location: `conf\upgrade10to1001` directory under your Unica Platform installation • Purpose: Import the **Unica Platform | Security | API Management | Unica Campaign** configuration node	configTool.bat -vp -p "Affinium|suite|security|apiSecurity|campaign" -f <Platform_Home>\conf\upgrade10to1001\APISecurity_triggered-Messages.xml
• File: `supportServer_config.xml` • Location: `conf\upgrade10to101` directory under your Unica Platform installation • Purpose: Import the **HCL Unica** configuration nodes	configTool.bat -vp -p "Affinium" -f <Platform_Home>\conf\upgrade10to101\supportServer_config.xml
• File: `cognos11_integration.xml` • Location: `conf\upgrade101to11` directory under your Unica Platform installation • Purpose: Import the **Affinium|Report|integrations|cognos11** configuration node	configTool.bat -vp -p "Affinium|Report|integrations" -f <Platform_Home>/conf/upgrade101to11/cognos11_integration.xml
• File: `watsonAssistant_naviga-tion.xml` • Location: `<PLATFORM_-HOME>\conf\watsonAssistan-t_navigation.xml` directory under your Unica Platform installation • Purpose: Import the `Affini-um|suite|uiNavigation|main-`	configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f "<PLATFORM_HOME>\conf\watsonAssistant_navigation.xml"

File details	Example command
`Menu|Analytics|`Watson Assistantconfiguration node configuration node	
• File: APISecurity.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the `Unica Platform | Security | API Management | Unica Platform | License node`	`configTool -vp -p "Affinium|suite|security" -f "<PLATFORM_HOME>\conf\upgrade12to121\APISecurity.xml"`
• APISecurity_Deliver.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the `Unica Platform | Security | API Management | Unica Deliver configuration node`	`configTool.bat -v -i -p "Affinium|suite|security|apiSecurity" -f "<PLATFORM_HOME>\conf\upgrade12to121\APISecurity_Deliver.xml"`
• LicenseDetailsNavigation.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the License details navigation option under the Settings menu	`configTool.bat -vp -p "Affinium| suite|uiNavigation|settingsMenu" -f "<PLATFORM_HOME>\conf\upgrade12to121\LicenseDeatilsNavigation.xml"`

File details	Example command
• File: APISecurity_GetInstalledApps.xml • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import new API configuration	`configTool.bat -vp -p "Affinium|suite|security|apiSecurity|manager" -f APISecurity_GetInstalledApps.xml`
• File: PlanJourneyIntegrationOption.xml • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import Plan Journey integration property	`configTool.bat -vp -p "Affinium|suite" -f PlanJourneyIntegrationOption.xml`
• File: `SiteMinder_AdditionalHeaderVariables.xml` • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import new property to accept additional header variables for Siteminder	`configTool.bat -vp -p "Affinium|suite|security|loginModes|siteMinderPartitionLogin" -f SiteMinder_AdditionalHeaderVariables.xml`
• File: `PasswordRules.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12103to12104` • Purpose: Import new password rules	`configTool.bat -vp -p "Affinium|Manager" -f PasswordRules.xml`

File details	Example command
• File: `support_Collaborate.xml` • Location: `<PLATFOR­M_HOME>\conf \upgrade12103to12104` • Purpose: Update CollaborateSupportServer property with new value	`configTool.bat -i -o -p "Affinium|suite" -f support_Collaborate.xml`
• File: `LookerIntegration.xml` • Location: `<PLATFOR­M_HOME>\conf \upgrade12104to1211` • Purpose: Import Google Looker integration configuration	`configtool.bat -vp -p "Affinium|Report|integrations" -f LookerIntegration.xml`
• File: `OpenInsights_navigation.xml` • Location: `<PLATFOR­M_HOME>\conf \upgrade12104to1211` • Purpose: Import navigation configuration for Open Insights menu	`configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f OpenInsights_navigation.xml`
• File: `OpenInsightsSeparator_navigation.xml` • Location: `<PLATFOR­M_HOME>\conf \upgrade12104to1211` • Purpose: Import configuration for adding separator for Open Insights menu	`configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f OpenInsightsSeparator_navigation.xml`

10. After you have imported the configuration properties in the previous step, restart the web application server on which Unica Platform is deployed so the changes are applied.

11. Complete the following steps to update the **Help > About** page.

    a. Use the **configTool** utility to export the **Affinium | Manager | about** category.

> ✏️ **Note:** The **Affinium | Manager | about** category is not visible on the **Configuration** page because it is marked hidden.

    Example (Windows™):

```
configTool.bat -x -p "Affinium|Manager|about" -f "C:\Unica\Platform
\conf\about.xml"
```

    b. Edit the exported XML file that you just created (`about.xml` in the example) to change the version number and display name.

    Find the `releaseNumber` property and change the value to the current version of Unica Platform. Change the value of the `copyright` property to the appropriate copyright year.

    In the following example, change the release number from 10.0.0.0.0 to 12.1.2.0.0 and change the copyright to 2021.

```
<property name="releaseNumber" type="string">

<displayNameKey>about.releaseNumber</displayNameKey>

<value>10.0.0.0.0.build_number</value>

</property>

</property>

<property id="541" name="copyright" type="string_property"

 width="40">

<value>2016</value>

</property>
```

    c. Use the **configTool** utility to import the revised file.

You must use the `-o` option to overwrite the node. You must specify the parent node when you import.

Example (Windows™):

```
configTool.bat -vp -i -p "Affinium|Manager" -f "about.xml" -o
```

12. Use the quartzjobtool to update scheduler jobs. This is a required step. If this upgrade tool is not run, any existing scheduled job fails 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`

13. Deploy and verify your installation as described in the Deploying Unica Platform *(on page 168)*.

Chapter 10. Upgrading from version 10.1.0 with manual migration

The Unica Platform upgrade installer can perform all of the data migration that is required for an upgrade automatically. However, if the policy of your organization does not allow automatic migration, you must complete the migration procedure to upgrade Unica Platform manually.

Ensure that you have the following installers in the same directory.

- The Unica master installer
- The Unica Platform installer

To upgrade to Unica Platform version 12.1.2 manually, you must run SQL scripts and run some command line utilities to populate the system tables. Ensure that your installation of Unica Platform is fully functional so that you can run the utilities. Complete information about using these utilities, including example commands for common tasks, is available in the following topics.

- populateDb *(on page 198)*
- configTool *(on page 184)*

The utilities are in the `tools\bin` directory under your Unica Platform installation.

See Unica Platform upgrade scenarios (on page 23) for information about upgrading from other versions.

1. Create a back-up copy of your Unica Platform system table database.

   ⚠ **Important:** Do not skip this step. If the upgrade operation fails, you cannot roll back your database and your data is corrupted.

2. Undeploy your Unica Platform deployment.
3. Run the Unica master installer.

The Unica master installer starts. Use the following instructions after theUnica master installer starts.

- When the Unica master installer prompts you to choose an installation directory, choose the root installation directory, not the Unica Platform installation directory that is under this root directory.
- When the Unica master installer prompts you to enter Unica Platform database connection information, enter the information that pertains to your current Unica Platform system tables.

The Unica master installer pauses and starts the Unica Platform installer.

4. Complete the following steps while the Unica Platform installer is running.

   a. When the Unica Platform installer prompts you for an installation directory, select the directory of your current Unica Platform installation, usually named `Platform`.

   b. If Contact Central is selected in component selection during installation, select database type.

   > **Note:** Contact Central is supported with OneDb and Oracle database.

   c. Provide Contact Central database connection details. Notes. By default only automatic database setup wil be executed for Control Center.

   d. Allow the installer to create a back-up copy of your previous installation of Unica Platform.

   e. Select **Manual database setup**.

   f. Clear the **Run Platform configuration** check box.

   g. Follow the remaining steps in the installation wizard and enter the requested information.

5. Run the following scripts against your system tables, where *DB_Type* is the database type.

Table 20. SQL scripts for upgrades from version 10.1.0

File details	File location
`ManagerSchema_DB_Type_11upg.sql`	`db\upgrade101to11` directory under your Unica Platform installation
`ManagerSchema_DB_Type_111upg.sql`	`db\upgrade1101to111` directory under your Unica Platform installation
`ManagerSchema_DB_Type_11102upg.sql`	`db\upgrade1111to11102 directory` under your Unica Platform installation
`ManagerSchema_DB_Type_12upg.sql`	`db\upgrade1102to12 directory` under your Unica Platform installation
`ManagerSchema_DB_Type_121upg.sql`	`db\upgrade12to121` directory under your Unica Platform installation
`ManagerUpdate_12101upg.sql`	`db\upgrade121to12101` directory under your Unica Platform installation
`ManagerUpdate_12103_DB_Type_upg.sql`	`db\upgrade12101to12103` directory under your Unica Platform installation
`ManagerUpdate_12104_DB_Type_upg.sql` and `DB_Type_QRTZ_Scheduler_12104_-upgrade_Script.sql`	`db\upgrade12103to12104` directory under your Unica Platform installation
`ManagerUpdate_1211_DB_Type_upg.sql`	`db\upgrade12104to1211` directory under your Unica Platform installation
`ManagerUpdate_1212_DB_Type_upg.sql` and `ContentIntegrationUpdate_1212_-DB_Type_upg.sql`	`db\upgrade1211to1212` directory under your Unica Platform installation

6. Use the `populateDb` utility to populate the system tables with default Unica Platform configuration properties, users and groups, and security roles and permissions.

The `populateDb` utility is in the `tools\bin` directory under your Unica Platform installation. Example command: `populateDb -n Manager`

7. Use the configTool utility to import configuration properties as described below.

⚠️ **Important:** Perform the imports in the order shown in the following table.

Table 21. Configuration properties for upgrades from version 10.1.0

File details	Example command
• File: `cognos11_integration.xml` • Location: `conf\upgrade101to11` directory under your Unica Platform installation • Purpose: Import the **Affinium|Report|integrations|cognos11** configuration node	configTool.bat -vp -p "Affinium|Report|integrations" -f <Platform_Home/conf/upgrade101to11/cognos11_integration.xml
• File: `watsonAssistant_navigation.xml` • Location: `<PLATFORM_-HOME>\conf\watsonAssistant_navigation.xml` directory under your Unica Platform installation • Purpose: Import the `Affinium|suite|uiNavigation|mainMenu|Analytics|`Watson Assistantconfiguration node configuration node	configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f "<PLATFORM_HOME>\conf\watsonAssistant_navigation.xml"
• File: APISecurity.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121`	`configTool -vp -p "Affinium|suite|security" -f "<PLATFORM_HOME>\conf \upgrade12to121\APISecurity.xml"`

File details	Example command
• Purpose: Import the `Unica Platform | Security | API Management | Unica Platform | License` node	
• APISecurity_Deliver.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the `Unica Platform | Security | API Management | Unica Deliver configuration` node	`configTool.bat -v -i -p "Affinium|suite|security|apiSecurity" -f "<PLATFORM_HOME>\conf\upgrade12to121\APISecurity_Deliver.xml"`
• LicenseDetailsNavigation.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the License details navigation option under the Settings menu	`configTool.bat -vp -p "Affinium| suite|uiNavigation|settingsMenu" -f "<PLATFORM_HOME>\conf\upgrade12to121\LicenseDeatilsNavigation.xml"`
• File: APISecurity_GetInstalledApps.xml • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import new API configuration	`configTool.bat -vp -p "Affinium|suite|security|apiSecurity|manager" -f APISecurity_GetInstalledApps.xml`

File details	Example command
• File: PlanJourneyIntegrationOption.xml • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import Plan Journey integration property	`configTool.bat -vp -p "Affinium|suite" -f PlanJourneyIntegrationOption.xml`
• File: `SiteMinder_AdditionalHeaderVariables.xml` • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import new property to accept additional header variables for Siteminder	`configTool.bat -vp -p "Affinium|suite|security|loginModes|siteMinderPartitionLogin" -f SiteMinder_AdditionalHeaderVariables.xml`
• File: `PasswordRules.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12103to12104` • Purpose: Import new password rules	`configTool.bat -vp -p "Affinium|Manager" -f PasswordRules.xml`
• File: `support_Collaborate.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12103to12104` • Purpose: Update CollaborateSupportServer property with new value	`configTool.bat -i -o -p "Affinium|suite" -f support_Collaborate.xml`

File details	Example command
• File: `LookerIntegration.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12104to1211` • Purpose: Import Google Looker integration configuration	`configtool.bat -vp -p "Affinium|Report|integrations" -f LookerIntegration.xml`
• File: `OpenInsights_navigation.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12104to1211` • Purpose: Import navigation configuration for Open Insights menu	`configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f OpenInsights_navigation.xml`
• File: `OpenInsightsSeparator_navigation.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12104to1211` • Purpose: Import configuration for adding separator for Open Insights menu	`configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f OpenInsightsSeparator_-navigation.xml`

8. After you have imported the configuration properties in the previous step, restart the web application server on which Unica Platform is deployed so the changes are applied.

9. Complete the following steps to update the **Help > About** page.

   a. Use the **configTool** utility to export the **Affinium | Manager | about** category.

   > **Note:** The **Affinium | Manager | about** category is not visible on the **Configuration** page because it is marked hidden.

Example (Windows™):

```
configTool.bat -x -p "Affinium|Manager|about" -f "C:\Unica\Platform
\conf\about.xml"
```

b. Edit the exported XML file that you just created (`about.xml` in the example) to change the version number and display name.

Find the `releaseNumber` property and change the value to the current version of Unica Platform. Change the value of the `copyright` property to the appropriate copyright year.

In the following example, change the release number from 10.1.0.0.0 to 12.1.2.0.0 and change the copyright to 2021.

```
<property name="releaseNumber" type="string">

<displayNameKey>about.releaseNumber</displayNameKey>

<value>10.1.0.0.0.build_number</value>

</property>

</property>

<property id="541" name="copyright" type="string_property"

 width="40">

<value>2016</value>

</property>
```

c. Use the **configTool** utility to import the revised file.

You must use the `-o` option to overwrite the node. You must specify the parent node when you import.

Example (Windows™):

```
configTool.bat -vp -i -p "Affinium|Manager" -f "about.xml" -o
```

10. Use the quartzjobtool to update scheduler jobs. This is a required step. If this upgrade tool is not run, any existing scheduled job fails 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`

11. Deploy and verify your installation as described in the Deploying Unica Platform *(on
page 168)*.

Chapter 11. Upgrading from version 11.0.0 with manual migration

The Unica Platform upgrade installer can perform all of the data migration that is required for an upgrade automatically. However, if the policy of your organization does not allow automatic migration, you must complete the migration procedure to upgrade Unica Platform manually.

Ensure that you have the following installers in the same directory.

- The Unica master installer
- The Unica Platform installer

To upgrade to Unica Platform version 12.1.2 manually, you must run SQL scripts and run some command line utilities to populate the system tables. Ensure that your installation of Unica Platform is fully functional so that you can run the utilities. Complete information about using these utilities, including example commands for common tasks, is available in the following topics.

- populateDb *(on page 198)*
- configTool *(on page 184)*

The utilities are in the `tools\bin` directory under your Unica Platform installation.

See Unica Platform upgrade scenarios (on page 23) for information about upgrading from other versions.

1. Create a back up copy of your Unica Platform system table database.

   ⚠️ **Important:** Do not skip this step. If the upgrade operation fails, you cannot roll back your database and your data is corrupted.

2. Undeploy your Unica Platform deployment.
3. Run the Unica master installer.

The Unica master installer starts. Use the following instructions after theUnica master installer starts.

- When the Unica master installer prompts you to choose an installation directory, choose the root installation directory, not the Unica Platform installation directory that is under this root directory.
- When the Unica master installer prompts you to enter Unica Platform database connection information, enter the information that pertains to your current Unica Platform system tables.

The Unica master installer pauses and starts the Unica Platform installer.

4. Complete the following steps while the Unica Platform installer is running.

    a. When the Unica Platform installer prompts you for an installation directory, select the directory of your current Unica Platform installation, usually named `Platform`.

    b. If Contact Central is selected in component selection during installation, select database type.

        📝 **Note:** Contact Central is supported with OneDb and Oracle database.

    c. Provide Contact Central database connection details. Notes. By default, only automatic database setup will be executed for Control Center.

    d. Allow the installer to create a back-up copy of your previous installation of Unica Platform.

    e. Select **Manual database setup**.

    f. Clear the **Run Platform configuration** check box.

    g. Follow the remaining steps in the installation wizard and enter the requested information.

5. Run the following scripts against your system tables, where *DB_Type* is the database type.

Table 22. SQL scripts for upgrades from version 11.0.0

File details	File location
`ManagerSchema_DB_Type_111upg.sql`	`db\upgrade1101to111` directory under your Unica Platform installation
`ManagerSchema_DB_Type_11102upg.sql`	`db\upgrade1111to11102 directory` under your Unica Platform installation
`ManagerSchema_DB_Type_12upg.sql`	`db\upgrade1102to12 directory` under your Unica Platform installation
`ManagerSchema_DB_Type_121upg.sql`	`db\upgrade12to121` directory under your Unica Platform installation
`ManagerUpdate_12101upg.sql`	`db\upgrade121to12101` directory under your Unica Platform installation
`ManagerUpdate_12103_DB_Type_upg.sql`	`db\upgrade12101to12103` directory under your Unica Platform installation
`ManagerUpdate_12104_DB_Type_upg.sql` and `DB_Type_QRTZ_Scheduler_12104_-upgrade_Script.sql`	`db\upgrade12103to12104` directory under your Unica Platform installation
`ManagerUpdate_1211_DB_Type_upg.sql`	`db\upgrade12104to1211` directory under your Unica Platform installation
`ManagerUpdate_1212_DB_Type_upg.sql` and `ContentIntegrationUpdate_1212_-DB_Type_upg.sql`	`db\upgrade1211to1212`directory under your Unica Platform installation

6. Use the `populateDb` utility to populate the system tables with default Unica Platform configuration properties, users and groups, and security roles and permissions.

   The `populateDb` utility is in the `tools\bin` directory under your Unica Platform installation. Example command: `populateDb -n Manager`

7. Use the configTool utility to import configuration properties as described below.

⚠ **Important:** Perform the imports in the order shown in the following table.

Table 23. Configuration properties for upgrades from version 11.0.0

File details	Example command
• File: `watsonAssistant_navigation.xml` • Location: `<PLATFORM_HOME>\conf\watsonAssistant_navigation.xml` directory under your Unica Platform installation • Purpose: Import the `Affinium|suite|uiNavigation|mainMenu|Analytics|`Watson Assistantconfiguration node configuration node	configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f "<PLATFORM_HOME>\conf\watsonAssistant_navigation.xml"
• File: APISecurity.xml • Location: `<PLATFORM_HOME>\conf\upgrade12to121` • Purpose: Import the `Unica Platform | Security | API Management | Unica Platform | License node`	```configTool -vp -p "Affinium|suite|security" -f "<PLATFORM_HOME>\conf\upgrade12to121\APISecurity.xml"```
• APISecurity_Deliver.xml • Location: `<PLATFORM_HOME>\conf\upgrade12to121` • Purpose: Import the `Unica Platform | Security | API Management | Unica Deliver configuration node`	```configTool.bat -v -i -p "Affinium|suite|security|apiSecurity" -f "<PLATFORM_HOME>\conf\upgrade12to121\APISecurity_Deliver.xml"```

File details	Example command
• LicenseDetailsNavigation.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the License details navigation option under the Settings menu	```configTool.bat -vp -p "Affinium| suite|uiNavigation|settings-Menu" -f "<PLATFORM_HOME>\conf\upgrade12to121\LicenseDeatilsNavigation.xml"```
• File: APISecurity_GetInstalledApps.xml • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import new API configuration	```configTool.bat -vp -p "Affinium|suite|security|apiSecurity|manager" -f APISecurity_GetInstalledApps.xml```
• File: PlanJourneyIntegrationOption.xml • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import Plan Journey integration property	```configTool.bat -vp -p "Affinium|suite" -f PlanJourneyIntegrationOption.xml```
• File: `SiteMinder_AdditionalHeaderVariables.xml` • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import new property to accept additional header variables for Siteminder	```configTool.bat -vp -p "Affinium|suite|security|loginModes|siteMinderPartitionLogin" -f SiteMinder_AdditionalHeaderVariables.xml```

File details	Example command
• File: `PasswordRules.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade12103to12104` • Purpose: Import new password rules	`configTool.bat -vp -p "Affini-um|Manager" -f PasswordRules.xml`
• File: `support_Collaborate.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade12103to12104` • Purpose: Update CollaborateSup-portServer property with new value	`configTool.bat -i -o -p "Affini-um|suite" -f support_Collabo-rate.xml`
• File: `LookerIntegration.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade12104to1211` • Purpose: Import Google Looker in-tegration configuration	`configtool.bat -vp -p "Affinium|Re-port|integrations" -f LookerInte-gration.xml`
• File: `OpenInsights_naviga-tion.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade12104to1211` • Purpose: Import navigation config-uration for Open Insights menu	`configtool.bat -vp -p "Affini-um|suite|uiNavigation|mainMenu|An-alytics" -f OpenInsights_naviga-tion.xml`

File details	Example command
• File: `OpenInsightsSeparator_nav-igation.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade12104to1211` • Purpose: Import configuration for adding separator for Open Insights menu	`configtool.bat -vp -p "Affini-um|suite|uiNavigation|mainMenu|An-alytics" -f OpenInsightsSeparator_-navigation.xml`

8. After you have imported the configuration properties in the previous step, restart the web application server on which Unica Platform is deployed so the changes are applied.

9. Complete the following steps to update the **Help > About** page.

   a. Use the **configTool** utility to export the **Affinium | Manager | about** category.

   > 📝 **Note:** The **Affinium | Manager | about** category is not visible on the **Configuration** page because it is marked hidden.

   Example (Windows™):

   ```
   configTool.bat -x -p "Affinium|Manager|about" -f "C:\Unica\Platform
   \conf\about.xml"
   ```

   b. Edit the exported XML file that you just created (`about.xml` in the example) to change the version number and display name.

   Find the `releaseNumber` property and change the value to the current version of Unica Platform. Change the value of the `copyright` property to the appropriate copyright year.

   As in the following example, change the release number from 11.0.0.0.0 to 12.1.2.0.0 and change the copyright to 2021.

```
<property name="releaseNumber" type="string">

<displayNameKey>about.releaseNumber</displayNameKey>

<value>11.0.0.0.0.build_number</value>

</property>

</property>

<property id="541" name="copyright" type="string_property"

 width="40">

<value>2016</value>

</property>
```

c. Use the **configTool** utility to import the revised file.

You must use the `-o` option to overwrite the node. You must specify the parent node when you import.

Example (Windows™):

```
configTool.bat -vp -i -p "Affinium|Manager" -f "about.xml" -o
```

10. Use the quartzjobtool to update scheduler jobs. This is a required step. If this upgrade tool is not run, any existing scheduled job fails 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

11. Deploy and verify your installation as described in the Deploying Unica Platform *(on page 168)*.

Chapter 12. Upgrading from version 11.0.1 with manual migration

The Unica Platform upgrade installer can perform all of the data migration that is required for an upgrade automatically. However, if the policy of your organization does not allow automatic migration, you must complete the migration procedure to upgrade Unica Platform manually.

Ensure that you have the following installers in the same directory.

- The Unica master installer
- The Unica Platform installer

To upgrade to Unica Platform version 12.1.2 manually, you must run SQL scripts and run some command line utilities to populate the system tables. Ensure that your installation of Unica Platform is fully functional so that you can run the utilities. Complete information about using these utilities, including example commands for common tasks, is available in the following topics.

- populateDb *(on page 198)*
- configTool *(on page 184)*

The utilities are in the `tools\bin` directory under your Unica Platform installation.

See Unica Platform upgrade scenarios (on page 23) for information about upgrading from other versions.

1. Create a back up copy of your Unica Platform system table database.

   ⚠️ **Important:** Do not skip this step. If the upgrade operation fails, you cannot roll back your database and your data is corrupted.

2. Undeploy your Unica Platform deployment.
3. Run the Unica master installer.

The Unica master installer starts. Use the following instructions after theUnica master installer starts.

- When the Unica master installer prompts you to choose an installation directory, choose the root installation directory, not the Unica Platform installation directory that is under this root directory.
- When the Unica master installer prompts you to enter Unica Platform database connection information, enter the information that pertains to your current Unica Platform system tables.

The Unica master installer pauses and starts the Unica Platform installer.

4. Complete the following steps while the Unica Platform installer is running.

   a. When the Unica Platform installer prompts you for an installation directory, select the directory of your current Unica Platform installation, usually named `Platform`.

   b. If Contact Central is selected in component selection during installation, select database type.

      📝 **Note:** Contact Central is supported with OneDb and Oracle database.

   c. Provide Contact Central database connection details. Notes. By default only automatic database setup will be executed for Control Center.

   d. Allow the installer to create a back-up copy of your previous installation of Unica Platform.

   e. Select **Manual database setup**.

   f. Clear the **Run Platform configuration** check box.

   g. Follow the remaining steps in the installation wizard and enter the requested information.

5. Run the following scripts against your system tables, where *DB_Type* is the database type.

Table 24. SQL scripts for upgrades from version 11.0.1

File details	File location
`ManagerSchema_DB_Type_111upg.sql`	`db\upgrade1101to111` directory under your Unica Platform installation
`ManagerSchema_DB_Type_11102upg.sql`	`db\upgrade1111to11102 directory` under your Unica Platform installation
`ManagerSchema_DB_Type_12upg.sql`	`db\upgrade1102to12 directory` under your Unica Platform installation
`ManagerSchema_DB_Type_121upg.sql`	`db\upgrade12to121` directory under your Unica Platform installation
`ManagerUpdate_12101upg.sql`	`db\upgrade121to12101` directory under your Unica Platform installation
`ManagerUpdate_12103_DB_Type_upg.sql`	`db\upgrade12101to12103` directory under your Unica Platform installation
`ManagerUpdate_12104_DB_Type_upg.sql` and `DB_Type_QRTZ_Scheduler_12104_-upgrade_Script.sql`	`db\upgrade12103to12104` directory under your Unica Platform installation
`ManagerUpdate_1211_DB_Type_upg.sql`	`db\upgrade12104to1211` directory under your Unica Platform installation
`ManagerUpdate_1212_DB_Type_upg.sql` and `ContentIntegrationUpdate_1212_-DB_Type_upg.sql`	`db\upgrade1211to1212`directory under your Unica Platform installation

6. Use the `populateDb` utility to populate the system tables with default Unica Platform configuration properties, users and groups, and security roles and permissions.

   The `populateDb` utility is in the `tools\bin` directory under your Unica Platform installation. Example command: `populateDb -n Manager`

7. Use the configTool utility to import configuration properties as described below.

⚠️ **Important:** Perform the imports in the order shown in the following table.

Table 25. Configuration properties for upgrades from version 11.1.0

File details	Example command
• File: `watsonAssistant_naviga-tion.xml` • Location: `<PLATFORM_-HOME>\conf\watsonAssistan-t_navigation.xml` directory under your Unica Platform installation • Purpose: Import the `Affini-um|suite|uiNavigation|main-Menu|Analytics|`Watson Assistantconfiguration node configuration node	configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f "<PLATFORM_HOME>\conf\watsonAssistant_navigation.xml"
| • File: APISecurity.xml<br>• Location: `<PLATFORM_-HOME>\conf\upgrade12to121`<br>• Purpose: Import the `Unica Plat-form \| Security \| API Manage-ment \| Unica Platform \| Li-cense node` | ```
configTool -vp -p "Affinium|suite|
security" -f "<PLATFORM_HOME>\conf
\upgrade12to121\APISecurity.xml"
``` |
| • APISecurity_Deliver.xml<br>• Location: `<PLATFORM_-HOME>\conf\upgrade12to121`<br>• Purpose: Import the `Unica Plat-form \| Security \| API Manage-ment \| Unica Deliver configu-ration node` | ```
configTool.bat -v -i -p "Affini-
um|suite|security|apiSecuri-
ty" -f "<PLATFORM_HOME>\conf\up-
grade12to121\APISecurity_Deliv-
er.xml"
``` |

File details	Example command
• LicenseDetailsNavigation.xml • Location: `<PLATFORM_-HOME>\conf\upgrade12to121` • Purpose: Import the License details navigation option under the Settings menu	`configTool.bat -vp -p "Affinium| suite|uiNavigation|settings-Menu" -f "<PLATFORM_HOME>\conf\upgrade12to121\LicenseDeatilsNavigation.xml"`
• File: APISecurity_GetInstalledApps.xml • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import new API configuration	`configTool.bat -vp -p "Affinium|suite|security|apiSecurity|manager" -f APISecurity_GetInstalledApps.xml`
• File: PlanJourneyIntegrationOption.xml • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import Plan Journey integration property	`configTool.bat -vp -p "Affinium|suite" -f PlanJourneyIntegrationOption.xml`
• File: `SiteMinder_AdditionalHeaderVariables.xml` • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import new property to accept additional header variables for Siteminder	`configTool.bat -vp -p "Affinium|suite|security|loginModes|siteMinderPartitionLogin" -f SiteMinder_AdditionalHeaderVariables.xml`

File details	Example command
• File: `PasswordRules.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12103to12104` • Purpose: Import new password rules	`configTool.bat -vp -p "Affinium|Manager" -f PasswordRules.xml`
• File: `support_Collaborate.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12103to12104` • Purpose: Update CollaborateSupportServer property with new value	`configTool.bat -i -o -p "Affinium|suite" -f support_Collaborate.xml`
• File: `LookerIntegration.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12104to1211` • Purpose: Import Google Looker integration configuration	`configtool.bat -vp -p "Affinium|Report|integrations" -f LookerIntegration.xml`
• File: `OpenInsights_navigation.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12104to1211` • Purpose: Import navigation configuration for Open Insights menu	`configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f OpenInsights_navigation.xml`

File details	Example command
• File: `OpenInsightsSeparator_navigation.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12104to1211` • Purpose: Import configuration for adding separator for Open Insights menu	`configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f OpenInsightsSeparator_navigation.xml`

8. After you have imported the configuration properties in the previous step, restart the web application server on which Unica Platform is deployed so the changes are applied.

9. Complete the following steps to update the **Help > About** page.

   a. Use the **configTool** utility to export the **Affinium | Manager | about** category.

      > **Note:** The **Affinium | Manager | about** category is not visible on the **Configuration** page because it is marked hidden.

      Example (Windows™):

      ```
      configTool.bat -x -p "Affinium|Manager|about" -f "C:\Unica\Platform\conf\about.xml"
      ```

   b. Edit the exported XML file that you just created (`about.xml` in the example) to change the version number and display name.

      Find the `releaseNumber` property and change the value to the current version of Unica Platform. Change the value of the `copyright` property to the appropriate copyright year.

      As in the following example, change the release number from 11.0.0.0.0 to 12.1.2.0.0 and change the copyright to 2021

```
<property name="releaseNumber" type="string">

<displayNameKey>about.releaseNumber</displayNameKey>

<value>11.0.0.0.0.build_number</value>

</property>

</property>

<property id="541" name="copyright" type="string_property"
 width="40">

<value>2016</value>

</property>
```

c. Use the **configTool** utility to import the revised file.

You must use the `-o` option to overwrite the node. You must specify the parent node when you import.

Example (Windows™):

```
configTool.bat -vp -i -p "Affinium|Manager" -f "about.xml" -o
```

10. Use the quartzjobtool to update scheduler jobs. This is a required step. If this upgrade tool is not run, any existing scheduled job fails 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

11. Deploy and verify your installation as described in the Deploying Unica Platform *(on page 168)*.

Chapter 13. Upgrading from version 11.1.0 with manual migration

The Unica Platform upgrade installer performs the data migration required for an upgrade automatically. However, if the policy of your organization does not allow automatic migration, you must complete the migration procedure to upgrade Unica Platform manually.

Ensure that you have the following installers in the same directory.

- The Unica master installer
- The Unica Platform installer

To upgrade to Unica Platform version 12.1.2 manually, you must run SQL scripts and run some command line utilities to populate the system tables. Ensure that your installation of Unica Platform is fully functional so that you can run the utilities. Complete information about using these utilities, including example commands for common tasks, is available in the following topics.

- configTool *(on page 184)*
- populateDb *(on page 198)*

The utilities are in the `tools\bin` directory under your Unica Platform installation.

See Unica Platform upgrade scenarios *(on page 23)* for information about upgrading from other versions.

1. Create a back-up copy of your Unica Platform system table database.

   Important: Do not skip this step. If the upgrade operation fails, you cannot roll back your database and your data is corrupted.

2. Undeploy your Unica Platform deployment.
3. Run the Unica master installer.

   The Unica master installer starts. Use the following instructions after the Unica master installer starts.

- When the Unica master installer prompts you to choose an installation directory, choose the root installation directory, not the Unica Platform installation directory that is under this root directory.
- When the Unica master installer prompts you to enter Unica Platform database connection information, enter the information that pertains to your current Unica Platform system tables.

The Unica master installer pauses and starts the Unica Platform installer.

4. Complete the following steps while the Unica Platform installer is running.

   a. When the Unica Platform installer prompts you for an installation directory, select the directory of your current Unica Platform installation, usually named **Platform**.

   b. Allow the installer to create a back-up copy of your previous installation of Unica Platform.

   c. If Contact Central is selected in component selection during installation, select database type.

   > ✏️ **Note:**
   >
   > Contact Central is supported with OneDb and Oracle database.

   d. Provide Contact Central database connection details. By default only automatic database setup will be executed for Control Center.

   e. Select **Manual database setup**.

   f. Clear the **Run Platform configuration** check box.

   g. Follow the remaining steps in the installation wizard and enter the requested information.

5. . Run the following scripts against your system tables, where `DB_Type` is the database type.

File details	File columns
`ManagerSchema_DB_Type_11102upg.sql`	db\upgrade1111to11102 directory under your Unica Platform installation

File details	File columns
`ManagerSchema_DB_Type_12upg.sql`	db\upgrade1102to12 directory under your Unica Platform installation
`ManagerSchema_DB_Type_121upg.sql`	`db\upgrade12to121` directory under your Unica Platform installation
`ManagerUpdate_12101upg.sql`	`db\upgrade121to12101` directory under your Unica Platform installation
`ManagerUpdate_12103_DB_Type_up-g.sql`	`db\upgrade12101to12103` directory under your Unica Platform installation
`ManagerUpdate_12104_DB_Type_-upg.sql` and `DB_Type_QRTZ_Sched-uler_12104_upgrade_Script.sql`	`db\upgrade12103to12104` directory under your Unica Platform installation
`ManagerUpdate_1211_DB_Type_up-g.sql`	`db\upgrade12104to1211` directory under your Unica Platform installation
`ManagerUpdate_1212_DB_Type_up-g.sql` and `ContentIntegrationUp-date_1212_DB_Type_upg.sql`	`db\upgrade1211to1212`directory under your Unica Platform installation

6. Use the configTool utility to import configuration properties as described below.

   Important: Perform the imports in the order shown in the following table.

   Table 26. Configuration properties for upgrades from version 11.0.1

File details	Example command

• File: `watsonAssistant_naviga-tion.xml`
• Location: `<PLATFORM_-HOME>\conf\watsonAssis-tant_navigation.xml` direc-
	`configtool.bat -vp -p`
	`"Affinium|`
	`suite|uiNavigation|mainMenu|`
	`Analytics" -f`
	`"<PLATFORM_HOME>\conf`

File details	Example command
tory under your Unica Platform installation • Purpose: Import the `Affini-um|suite|uiNavigation|main-Menu|Analytics|Watson Assis-tant` configuration node	`\watsonAssistant_navigation.xml"`
• File: APISecurity.xml • Location: `<PLATFOR-M_HOME>\conf\up-grade12to121` • Purpose: Import the `Unica Platform | Security | API Management | Unica Platform | License` node	`configTool -vp -p "Affini-um|suite|security" -f "<PLATFORM_HOME>\conf\up-grade12to121\APISecurity.xml"`
• APISecurity_Deliver.xml • Location: `<PLATFOR-M_HOME>\conf\up-grade12to121` • Purpose: Import the `Unica Platform | Security | API Management | Unica Deliver` configuration node	`configTool.bat -v -i -p "Affini-um|suite|security|apiSecuri-ty" -f "<PLATFORM_HOME>\conf\up-grade12to121\APISecurity_Deliv-er.xml"`
• File: `LicenseDetailsNaviga-tion.xml` • Location: `<PLATFOR-M_HOME>\conf\up-grade12to121` • Purpose: Import the License details navigation option under the Settings menu	`configTool.bat -vp -p "Affini-um| suite|uiNavigation|settings-Menu" -f "<PLATFORM_HOME>\conf \upgrade12to121\LicenseDeatils-Navigation.xml"`

File details	Example command
• File: `APISecurity_GetIn-stalledApps.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade121to12103` • Purpose: Import new API config-uration	`configTool.bat -vp -p "Affini-um|suite|security|apiSecuri-ty|manager" -f APISecurity_GetIn-stalledApps.xml`
• File: `PlanJourneyIntegration-Option.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade121to12103` • Purpose: Import Plan Journey in-tegration property	`configTool.bat -vp -p "Affini-um|suite" -f PlanJourneyIntegra-tionOption.xml`
• File: `SiteMinder_Additional-HeaderVariables.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade121to12103` • Purpose: Import new property to accept additional header vari-ables for Siteminder	`configTool.bat -vp -p "Affini-um|suite|security|login-Modes|siteMinderPartitionLogin" -f SiteMinder_AdditionalHeader-Variables.xml`
• File: `PasswordRules.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade12103to12104` • Purpose: Import new password rules	`configTool.bat -vp -p "Affini-um|Manager" -f PasswordRules.xml`

File details	Example command
• File: `support_Collaborate.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12103to12104` • Purpose: Update CollaborateSupportServer property with new value	`configTool.bat -i -o -p "Affinium|suite" -f support_Collaborate.xml`
• File: `LookerIntegration.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12104to1211` • Purpose: Import Google Looker integration configuration	`configtool.bat -vp -p "Affinium|Report|integrations" -f LookerIntegration.xml`
• File: `OpenInsights_navigation.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12104to1211` • Purpose: Import navigation configuration for Open Insights menu	`configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f OpenInsights_navigation.xml`
• File: `OpenInsightsSeparator_navigation.xml` • Location: `<PLATFORM_HOME>\conf \upgrade12104to1211` • Purpose: Import configuration for adding separator for Open Insights menu	`configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f OpenInsightsSeparator_navigation.xml`

7. After you have imported the configuration properties in the previous step, restart the web application server on which Unica Platform is deployed so the changes are applied.

8. Complete the following steps to update the **Help > About** page.

   a. Use the configTool utility to export the `Affinium | Manager | about` category.

   > ✏️ **Note:** The `Affinium | Manager | about category` is not visible on the Configuration page because it is marked hidden.

   Example (Windows): `configTool.bat -x -p "Affinium|Manager|about" -f "C:\Unica\Platform\conf\about.xml"`

   b. Edit the exported XML file that you just created (about.xml in the example) to change the version number and display name.

   Find the releaseNumber property and change the value to the current version of Unica Platform. Change the value of the copyright property to theappropriate copyright year.

   As in the following example, change the release number from 11.1.0.0.0 to 12.1.2.0.0 and change the copyright to 2021.

   ```
   <property name="releaseNumber" type="string">

   <displayNameKey>about.releaseNumber</displayNameKey>

   <value>11.1.0.0.0.build_number</value>

   </property>

   </property>

   <property id="541" name="copyright" type="string_property"

   width="40">

   <value>2016</value>

   </property>
   ```

   c. Use the configTool utility to import the revised file.

You must use the `-o` option to overwrite the node. You must specify the parent node when you import.

Example (Windows): `configTool.bat -vp -i -p "Affinium|Manager" -f "about.xml" -o`

9. 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`

10. Deploy and verify your installation in the Deploying Unica Platform *(on page 168)* topic.

Chapter 14. Upgrading from version 12.0.0 with manual migration

The Unica Platform upgrade installer performs the data migration required for an upgrade automatically. However, if the policy of your organization does not allow automatic migration, you must complete the migration procedure to upgrade Unica Platform manually.

Ensure that you have the following installers in the same directory.

- The Unica master installer
- The Unica Platform installer

To upgrade to Unica Platform version 12.1.2 manually, you must run SQL scripts and run some command line utilities to populate the system tables. Ensure that your installation of Unica Platform is fully functional so that you can run the utilities. Complete information about using these utilities, including example commands for common tasks, is available in the following topics.

- configTool *(on page 184)*
- populateDb *(on page 198)*

The utilities are in the `tools\bin` directory under your Unica Platform installation.

See Unica Platform upgrade scenarios *(on page 23)* for information about upgrading from other versions.

1. Create a back-up copy of your Unica Platform system table database.

   Important: Do not skip this step. If the upgrade operation fails, you cannot roll back your database and your data is corrupted.
2. Undeploy your Unica Platform deployment.
3. Run the Unica master installer.

   The Unica master installer starts. Use the following instructions after the Unica master installer starts.

- When the Unica master installer prompts you to choose an installation directory, choose the root installation directory, not the Unica Platform installation directory that is under this root directory.
- When the Unica master installer prompts you to enter Unica Platform database connection information, enter the information that pertains to your current Unica Platform system tables.

The Unica master installer pauses and starts the Unica Platform installer.

4. Complete the following steps while the Unica Platform installer is running.
   a. When the Unica Platform installer prompts you for an installation directory, select the directory of your current Unica Platform installation, usually named **Platform**.
   b. Allow the installer to create a back-up copy of your previous installation of Unica Platform.
   c. If Contact Central is selected in component selection during installation, select database type.

   📝 **Note:**

   Contact Central is supported with OneDb and Oracle database.

   d. Provide Contact Central database connection details. By default only automatic database setup will be executed for Control Center.
   e. Select **Manual database setup**.
   f. Clear the **Run Platform configuration** check box.
   g. Follow the remaining steps in the installation wizard and enter the requested information.

5. Run the following scripts against your system tables, where `DB_Type` is the database type.

File details	File columns
`ManagerSchema_DB_Type_121upg.sql`	`db\upgrade12to121` directory under your

	Unica Platform installation
`ManagerUpdate_12101upg.sql`	`db\upgrade121to12101` directory under your Unica Platform installation
`ManagerUpdate_12103_DB_Type_upg.sql`	`db\upgrade12101to12103` directory under your Unica Platform installation
`ManagerUpdate_12104_DB_Type_upg.sql` and `DB_Type_QRTZ_Scheduler_12104_upgrade_Script.sql`	`db\upgrade12103to12104` directory under your Unica Platform installation
`ManagerUpdate_1211_DB_Type_upg.sql`	`db\upgrade12104to1211` directory under your Unica Platform installation
`ManagerUpdate_1212_DB_Type_upg.sql` and `ContentIntegrationUpdate_1212_DB_-Type_upg.sql`	`db\upgrade1211to1212` directory under your Unica Platform installation

6. Use the configTool utility to import configuration properties as described below.

Important: Perform the imports in the order shown in the following table.

Table 27. Configuration properties for upgrades from version 12.0.0

File details	Example command
- File: `APISecurity.xml` - Location: `<PLATFORM_HOME>\conf\upgrade12to121` - Purpose: Import the `Unica Platform | Security | API Management | Unica Platform | License node`	`configTool -vp -p "Affinium|suite|security" -f "<PLATFORM_HOME>\conf\upgrade12to121\APISecurity.xml"`

• File: `APISecurity_Deliver.xml` • Location: `<PLATFORM_HOME>\conf\upgrade12to121` • Purpose: Import the `Unica Platform | Security | API Management | Unica Deliver configuration node`	`configTool.bat -v -i -p "Affinium|suite|security|apiSecurity" -f "<PLATFORM_HOME>\conf\upgrade12to121\APISecurity_Deliver.xml"`
• File: `LicenseDetailsNavigation.xml` • Location: `<PLATFORM_HOME>\conf\upgrade12to121` • Purpose: Import the License details navigation option under the Settings menu	`configTool.bat -vp -p "Affinium| suite|uiNavigation|settingsMenu" -f "<PLATFORM_HOME>\conf\upgrade12to121\LicenseDeatilsNavigation.xml"`
• File: `APISecurity_GetInstalledApps.xml` • Location: `<PLATFORM_HOME>\conf \upgrade121to12103` • Purpose: Import new API configuration	`configTool.bat -vp -p "Affinium|suite| security|apiSecurity|manager" -f APISecurity_GetInstalledApps.xml`

• File: `PlanJourneyIntegra-tionOption.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade121to12103` • Purpose: Import Plan Jour-ney integration property	`configTool.bat -vp -p "Affinium|suite"` `-f PlanJourneyIntegrationOption.xml`

• File: `SiteMinder_Ad-ditionalHeaderVari-ables.xml`
• Location: `<PLATFOR-M_HOME>\conf \up-grade121to12103`
• Purpose: Import new prop-erty to accept additional header variables for Site-minder
	`configTool.bat -vp -p "Affinium|suite|`
	`security|loginModes|siteMinderParti-`
	`tionLogin" -f SiteMinder_Additional-`
	`HeaderVariables.xml`

• File: `PasswordRules.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade12103to12104` • Purpose: Import new pass-word rules	`configTool.bat -vp -p "Affinium|Manag-` `er" -f PasswordRules.xml`
• File: `support_Collabo-rate.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade12103to12104` • Purpose: Update Collabo-rateSupportServer property with new value	`configTool.bat -i -o -p "Affini-` `um|suite" -f support_Collaborate.xml`

• File: `LookerIntegra-tion.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade12104to1211` • Purpose: Import Google Looker integration configuration	`configtool.bat -vp -p "Affinium|Re-port|integrations" -f LookerIntegra-tion.xml`
• File: `OpenInsights_navi-gation.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade12104to1211` • Purpose: Import navigation configuration for Open Insights menu	`configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f OpenInsights_navigation.xml`
• File: `OpenInsightsSepara-tor_navigation.xml` • Location: `<PLATFOR-M_HOME>\conf \up-grade12104to1211` • Purpose: Import configuration for adding separator for Open Insights menu	`configtool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f OpenInsightsSeparator_navigation.xml`

7. After you have imported the configuration properties in the previous step, restart the web application server on which Unica Platform is deployed so the changes are applied.

8. Complete the following steps to update the **Help > About** page.

a. Use the configTool utility to export the `Affinium | Manager | about` category.

> **Note:** The `Affinium | Manager | about category` is not visible on the Configuration page because it is marked hidden.

Example (Windows): `configTool.bat -x -p "Affinium|Manager|about" -f "C:\Unica\Platform\conf\about.xml"`

b. Edit the exported XML file that you just created (about.xml in the example) to change the version number and display name.

Find the releaseNumber property and change the value to the current version of Unica Platform. Change the value of the copyright property to the appropriate copyright year.

As in the following example, change the release number from 11.1.0.0.0 to 12.1.2.0.0 and change the copyright to 2021.

```
<property name="releaseNumber" type="string">

<displayNameKey>about.releaseNumber</displayNameKey>

<value>11.1.0.0.0.build_number</value>

</property>

</property>

<property id="541" name="copyright" type="string_property"
width="40">

<value>2016</value>

</property>
```

c. Use the configTool utility to import the revised file.

You must use the `-o` option to overwrite the node. You must specify the parent node when you import.

Example (Windows): `configTool.bat -vp -i -p "Affinium|Manager" -f "about.xml" -o`

9. 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`

10. Deploy and verify your installation in the Deploying Unica Platform *(on page 168)* topic.

Chapter 15. Upgrading from version 12.1.0 with manual migration

The Unica Platform upgrade installer performs the data migration required for an upgrade automatically. However, if the policy of your organization does not allow automatic migration, you must complete the migration procedure to upgrade Unica Platform manually.

Ensure that you have the following installers in the same directory.

- The Unica master installer
- The Unica Platform installer

To upgrade to Unica Platform version 12.1.2 manually, you must run SQL scripts and run some command line utilities to populate the system tables. Ensure that your installation of Unica Platform is fully functional so that you can run the utilities. Complete information about using these utilities, including example commands for common tasks, is available in the following topics.

The utilities are in the `tools\bin` directory under your Unica Platform installation.

See for information about upgrading from other versions.

1. Create a back-up copy of your Unica Platform system table database.

   Important: Do not skip this step. If the upgrade operation fails, you cannot roll back your database and your data is corrupted.
2. Undeploy your Unica Platform deployment.
3. Run the Unica master installer.

   The Unica master installer starts. Use the following instructions after the Unica master installer starts.

- When the Unica master installer prompts you to choose an installation directory, choose the root installation directory, not the Unica Platform installation directory that is under this root directory.
- When the Unica master installer prompts you to enter Unica Platform database connection information, enter the information that pertains to your current Unica Platform system tables.

The Unica master installer pauses and starts the Unica Platform installer.

4. Complete the following steps while the Unica Platform installer is running.

   a. When the Unica Platform installer prompts you for an installation directory, select the directory of your current Unica Platform installation, usually named **Platform**.

   b. Allow the installer to create a back-up copy of your previous installation of Unica Platform.

   c. If Contact Central is selected in component selection during installation, select database type.

   > ✏️ **Note:**
   >
   > Contact Central is supported with OneDb and Oracle database.

   d. Provide Contact Central database connection details. By default only automatic database setup will be executed for Control Center.

   e. Select **Manual database setup**.

   f. Clear the **Run Platform configuration** check box.

   g. Follow the remaining steps in the installation wizard and enter the requested information.

5. Run the following scripts against your system tables, where `DB_Type` is the database type.

File details	File columns
`ManagerUpdate_12101upg.sql`	`db\upgrade121to12101` directory under your Unica Platform installation

`ManagerUpdate_12103_DB_Type_upg.sql`	`db\upgrade12101to12103` directory under your Unica Platform installation
`ManagerUpdate_12104_DB_Type_upg.sql` and `DB_Type_QRTZ_Scheduler_12104_upgrade_Script.sql`	`db\upgrade12103to12104` directory under your Unica Platform installation
`ManagerUpdate_1211_DB_Type_upg.sql`	`db\upgrade12104to1211` directory under your Unica Platform installation
`ManagerUpdate_1212_DB_Type_upg.sql` and `ContentIntegrationUpdate_1212_DB_Type_upg.sql`	`db\upgrade1211to1212`directory under your Unica Platform installation

6. Use the configTool utility to import configuration properties as described below.

   Important: Perform the imports in the order shown in the following table.

Table 28. Configuration properties for upgrades from version 12.1.0

File details	Example command
File: `APISecurity_GetInstalledApps.xml` Location: `<PLATFORM_HOME>\conf \upgrade121to12103` Purpose: Import new API configuration	`configTool.bat -vp -p "Affinium|suite|security|apiSecurity|manager" -f APISecurity_GetInstalledApps.xml`
File: `PlanJourneyIntegrationOption.xml` Location: `<PLATFORM_HOME>\conf \upgrade121to12103`	`configTool.bat -vp -p "Affinium|suite" -f PlanJourneyIntegrationOption.xml`

Purpose: Import Plan Journey integration property	
File: `SiteMinder_AdditionalHeaderVariables.xml` Location: `<PLATFORM_HOME>\conf \upgrade121to12103` Purpose: Import new property to accept additional header variables for Siteminder	`configTool.bat -vp -p "Affinium|suite|security|loginModes|siteMinderPartitionLogin" -f SiteMinder_AdditionalHeaderVariables.xml`
File: `PasswordRules.xml` Location: `<PLATFORM_HOME>\conf \upgrade12103to12104` Purpose: Import new password rules	`configTool.bat -vp -p "Affinium|Manager" -f PasswordRules.xml`
File: `support_Collaborate.xml` Location: `<PLATFORM_HOME>\conf \upgrade12103to12104` Purpose: Update CollaborateSupportServer property with new value.	`configTool.bat -i -o -p "Affinium|suite" -f support_Collaborate.xml`
File: `LookerIntegration.xml`	`configtool.bat -vp -p "Affinium|Report|integrations" -f LookerIntegration.xml`

Location: `<PLATFOR-M_HOME>\conf \up-grade12104to1211` Purpose: Import Google Looker integration configuration	
File: `OpenInsights_navigation.xml` Location: `<PLATFOR-M_HOME>\conf \up-grade12104to1211` Purpose: Import navigation configuration for Open Insights menu.	`configtool.bat -vp -p "Affinium|suite|ui-Navigation|mainMenu|Analytics" -f OpenInsights_navigation.xml`
File: `OpenInsightsSeparator_navigation.xml` Location: `<PLATFOR-M_HOME>\conf \up-grade12104to1211` Purpose: Import configuration for adding separator for Open Insights menu.	`configtool.bat -vp -p "Affinium|suite|ui-Navigation|mainMenu|Analytics" -f OpenInsightsSeparator_navigation.xml`

7. After you have imported the configuration properties in the previous step, restart the web application server on which Unica Platform is deployed so the changes are applied.

8. Complete the following steps to update the **Help > About** page.

a. Use the configTool utility to export the `Affinium | Manager | about` category.

> **Note:** The `Affinium | Manager | about category` is not visible on the Configuration page because it is marked hidden.

Example (Windows): `configTool.bat -x -p "Affinium|Manager|about" -f "C:\Unica\Platform\conf\about.xml"`

b. Edit the exported XML file that you just created (about.xml in the example) to change the version number and display name.

Find the releaseNumber property and change the value to the current version of Unica Platform. Change the value of the copyright property to theappropriate copyright year.

As in the following example, change the release number from 12.1.0.0.0 to 12.1.2.0.0 and change the copyright to 2021.

```
<property name="releaseNumber" type="string">
<displayNameKey>about.releaseNumber</displayNameKey>
<value>12.1.0.0.0.build_number</value>
</property>
</property>
<property id="541" name="copyright" type="string_property"
width="40">
<value>2016</value>
</property>
```

c. Use the configTool utility to import the revised file.

You must use the `-o` option to overwrite the node. You must specify the parent node when you import.

Example (Windows): `configTool.bat -vp -i -p "Affinium|Manager" -f "about.xml" -o`

9. 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`

10. Deploy and verify your installation in the Deploying Unica Platform *(on page 168)* topic.

Chapter 16. Upgrading from version 12.1.1 with manual migration

The Unica Platform upgrade installer performs the data migration required for an upgrade automatically. However, if the policy of your organization does not allow automatic migration, you must complete the migration procedure to upgrade Unica Platform manually.

Ensure that you have the following installers in the same directory.

- The Unica master installer
- The Unica Platform installer

To upgrade to Unica Platform version 12.1.2 manually, you must run SQL scripts and run some command line utilities to populate the system tables. Ensure that your installation of Unica Platform is fully functional so that you can run the utilities. Complete information about using these utilities, including example commands for common tasks, is available in the following topics.

- configTool *(on page 184)*
- populateDb *(on page 198)*

The utilities are in the `tools\bin` directory under your Unica Platform installation.

See Unica Platform upgrade scenarios *(on page 23)* for information about upgrading from other versions.

1. Create a back-up copy of your Unica Platform system table database.

   Important: Do not skip this step. If the upgrade operation fails, you cannot roll back your database and your data is corrupted.
2. Undeploy your Unica Platform deployment.
3. Run the Unica master installer.

   The Unica master installer starts. Use the following instructions after the Unica master installer starts.

- When the Unica master installer prompts you to choose an installation directory, choose the root installation directory, not the Unica Platform installation directory that is under this root directory.

- When the Unica master installer prompts you to enter Unica Platform database connection information, enter the information that pertains to your current Unica Platform system tables.

The Unica master installer pauses and starts the Unica Platform installer.

4. Complete the following steps while the Unica Platform installer is running.

   a. When the Unica Platform installer prompts you for an installation directory, select the directory of your current Unica Platform installation, usually named **Platform**.

   b. Allow the installer to create a back-up copy of your previous installation of Unica Platform.

   c. If Contact Central is selected in component selection during installation, select database type.

   ✏️ **Note:**

   Contact Central is supported with OneDb and Oracle database.

   d. Provide Contact Central database connection details. By default only automatic database setup will be executed for Control Center.

   e. Select **Manual database setup**.

   f. Clear the **Run Platform configuration** check box.

   g. Follow the remaining steps in the installation wizard and enter the requested information.

5. Run the following scripts against your system tables, where `DB_Type` is the database type.

File details	File columns
`ManagerUpdate_1212_DB_Type_upg.sql` and `ContentIntegrationUpdate_1212_DB_-Type_upg.sql`	`db\upgrade1211to1212`directory under your Unica Platform installation

6. There is no additional configuration required for this upgrade. Restart the web application server on which Unica Platform is deployed so the changes are applied.

7. Complete the following steps to update the **Help > About** page.

   a. Use the configTool utility to export the `Affinium | Manager | about` category.

   > 📝 **Note:** The `Affinium | Manager | about category` is not visible on the Configuration page because it is marked hidden.

   Example (Windows): `configTool.bat -x -p "Affinium|Manager|about" -f "C:\Unica\Platform\conf\about.xml"`

   b. Edit the exported XML file that you just created (about.xml in the example) to change the version number and display name.

   Find the releaseNumber property and change the value to the current version of Unica Platform. Change the value of the copyright property to theappropriate copyright year.

   As in the following example, change the release number from 12.1.1.0.0 to 12.1.2.0.0 and change the copyright to 2021.

   ```
   <property name="releaseNumber" type="string">

   <displayNameKey>about.releaseNumber</displayNameKey>

   <value>12.1.1.0.0.build_number</value>

   </property>

   </property>

   <property id="541" name="copyright" type="string_property"
   width="40">

   <value>2016</value>

   </property>
   ```

   c. Use the configTool utility to import the revised file.

You must use the `-o` option to overwrite the node. You must specify the parent node when you import.

Example (Windows): `configTool.bat -vp -i -p "Affinium|Manager" -f "about.xml" -o`

8. 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`

9. Deploy and verify your installation in the Deploying Unica Platform *(on page 168)* topic.

Chapter 17. Deploying Unica Contact Central

Prerequisites

Unica Contact Central requires JRE 1.8 or higher to work.

> 📝 **Note:** Unica Contact Central does not support clustered environment.

ContactCentralDS JNDI resource for Contact Central database is required.

Setting Unica Contact Central and Unica Platform home JVM argument

Add the following JVM argument to the application server, where Unica Contact Central is required to be deployed.

`- DCONTACTCENTRAL_HOME = path_where_contact_central_is_installed`

`- DUNICA_PLATFORM_HOME = platform_installation_home_directory_path`

For example: `CONTACTCENTRAL_HOME =/opt/Platform/ContactCentral`

Point CONTACTCENTRAL_HOME to the directory, where Unica Contact Central is installed. It is installed within the ContactCentral directory under UNICA_PLATFORM_HOME.

Provide write permissions to the logs directory already created inside CONTACTCENTRAL_HOME.

Deployment procedure

You must follow a set of guidelines, when you deploy Unica Contact Central on your web application server When you ran the suite installer, you completed the following action:

- You created the WAR file of Unica Contact Central (ContactCentral.war) inside `UNICA_PLATFORM_HOME/ContactCentral` directory. UNICA_PLATFORM_HOME refers to the Unica Platform installation location.

✎ **Note:** It is assumed that you possess information on how to work with your web application server. For more information, see the web application server documentation.

Guidelines for deploying Unica Contact Central on WebSphere

You must follow a set of guidelines when you deploy Unica Contact Central onWebSphere. Ensure that the version of WebSphere meets the requirements that are described in the Recommended Software Environments and Minimum System Requirements document, including any required fix packs. Use the following guidelines when deploying Unica Contact Central on WebSphere:

1. Specify the following custom property in the server:

   Name: `com.ibm.ws.webcontainer.invokefilterscompatibility`

   Value: `true`

2. Deploy the ContactCentral.war file as an enterprise application. When you deploy the ContactCentral.war file, ensure that the JDK source level is set to 18 for SDK 1.8:
   a. In the form, select the WAR file, select **Show me all installation options** andparameters so the **Select Installation Options** wizard runs.
   b. In step 3 of the **Select Installation Options** wizard, ensure that the JDK SourceLevel is set to 18 for SDK 1.8.
   c. In step 8 of the **Select Installation Options** wizard, select ContactCentralDS as the matching Target Resource. ContactCentralDS JNDI resource must be present for Contact Central. Guidelines for setting up ContactCentralDS remains same as setting up UnicaPlatformDS datasource for Platform. For more information on setting up UnicaPlatformDS datasource, see *Unica Platform Installation Guide*.
   d. In step 10 of the **Select Installation Options** wizard, the context root must be set to /ContactCentral.

3. For your installation to support non-ASCII characters, for example for Portuguese or for locales that require multi-byte characters, add the following arguments to Generic JVMArguments at the server level.

-Dfile.encoding=UTF-8

-Dclient.encoding.override=UTF-8

> **Note:** Navigation tip: Select **Servers > Application Servers > Java and ProcessManagement > Process Definition > Java Virtual Machine > Generic JVM Arguments**.See the WebSphere documentation for additional details.

4. In the **Applications > Enterprise Applications** section of the server, select the WAR filethat you deployed, then
   - For Class loader order, select **Classes loaded with local class loader first (parentlast)**.
   - For WAR class loader policy, select **Single class loader for application**.
5. Start your deployment. If your instance of WebSphere is configured to use a JVMversion 1.8 or higher, complete the following steps to work around an issue with thetime zone database.
   a. Stop WebSphere.
   b. Download the Time Zone Update Utility for Java (JTZU).
   c. Follow the steps provided by the IBM (JTZU) to update the time zone data in yourJVM.
   d. Restart WebSphere.
6. In Websphere Enterprise Applications, select your **Application > Manage Modules >Your Application > Class Loader Order > Classes** loaded with local class loader first(parent last).
7. The recommended minimum heap size for the basic functioning of the application is 512 MB and the recommended maximum heap size is 1024 MB. Complete the following tasks to specify the heap size.

a. In WebSphere Enterprise Applications, select **Servers > WebSphere application servers > server1 > Server Infrastructure > Java and Process Management >Process definition > Java Virtual Machine**.

b. Set the initial heap size to 512m.

c. Set the maximum heap size to 1024m.

> ✎ **Note:** See the *WebSphere documentation* for more information about sizing.

For DB2, set progressiveStreaming = 2 in WebSphere console at following path: JDBC>Data sources > ContactCentralDS > Custom properties.

Guidelines for deploying Unica Contact Central on Apache Tomcat®

You must follow a set of guidelines when you deploy Unica Contact Central onApache Tomcat. Ensure that the version of Apache Tomcat meets the requirementsthat are described in the *Recommended Software Environments and Minimum SystemRequirements* document. Use the following guidelines when you deploy Unica ContentIntegration on Apache Tomcat.

1. Deploy the HCL ContactCentral.war file as an enterprise application on TomcatApache server. Complete the following tasks if your installation must support non-ASCIIcharacters, for example for Portuguese or for locales that require multi-byte characters.

2. Edit the setenv.sh file for the respective product instances script in the bindirectory under your tomcat instances directory to add `-Dfile.encoding=UTF-8Dclient.encoding.override=UTF-8`.

3. Restart Tomcat.

Unica Contact Central | User Role Creation

The procedure included in this section is not required for the following scenarios:

It is a fresh installation. The installation process creates this role under default partition(partition1).

It is an upgrade and only one partition exists in the current system.

In case, it is an upgrade and system contains more than one partition, you must execute the following command manually for each partition, except partition1. The command exists under <PLATFORM_HOME>\tools\bin directory inside your Platform installation directory. This creates a Contact Central user role under each partition.

On Windows

```
populateDb.bat -n ContactCentral -p <partition_name>
```

On Unix

```
populateDb.sh -n ContactCentral -p <partition_name>
```

Chapter 18. Deploying Unica Platform

You must follow a set of guidelines when you deploy Unica Platform in your web application server. There is a different set of guidelines for deploying Unica Platform on WebLogic and on WebSphere®.

When you ran the HCL installer, you completed one of the following actions:

- You included Unica Platform in an EAR file.
- You created the WAR file of Unica Platform (`unica.war`).

If you included other products in an EAR file, you must follow the deployment guidelines that are provided in the individual installation guides for the products that are included in the EAR file.

We assume that you know how to work with your web application server. Consult your web application server documentation for details such as navigation in the **Administration** console.

Important: HCL Unica products are license based, users are required to configure required licenses with HCL Unica products to start using them.

The license consumption will only be measured on Production setup.

Read carefully the HCL Unica Licensing Guide to understand how to configure, register license server and follow license consumption.

Guidelines for deploying Unica Platform on WebLogic

You must follow a set of guidelines when you deploy Unica Platform on the WebLogic application.

Use the following guidelines when you deploy the Unica Platform products on any supported version of WebLogic:

- Unica products customize the Java™ virtual machine (JVM) that is used by WebLogic. If you encounter errors related to JVM, you can create a WebLogic instance that is dedicated to Unica products.
- Open the `startWebLogic.cmd` file and verify that the SDK that is selected for the WebLogic domain that you are using is the Sun SDK for the **JAVA_VENDOR** variable.

  The **JAVA_VENDOR** variable must be set to `Sun` (**JAVA_VENDOR=Sun**). If the**JAVA_VENDOR** variable is set to **JAVA_VENDOR**, it means that JRockit is selected. You must change the selected SDK, because JRockit is not supported. Refer to the BEA WebLogic documentation to change the selected SDK.
- Deploy Unica Platform as a web application.
- If you are configuring WebLogic to use the IIS plug-in, review the BEA WebLogic documentation.
- Complete the following tasks if your installation must support non-ASCII characters, for example for Portuguese or for locales that require multi-byte characters:
  1. Edit the setDomainEnv script in the `bin` directory under your WebLogic domain directory to add the following Java options to `JAVA_VENDOR`:

     `-Dfile.encoding=UTF-8`

     In case you are deploying a non-production setup, add

     `-DENABLE_NON_PROD_MODE=true`

     In case of a production setup, this Java option must be removed or set to false.
  2. In the **WebLogic** console, click the **Domain** link on the home page.
  3. In the **Web Applications** tab, select the **Archived Real Path Enabled** check box.
  4. Restart WebLogic.
  5. Deploy and start the EAR file or the `unica.war` file.
- If deploying in a production environment, set the JVM memory heap size parameters to 1024 by adding the following line to the `setDomainEnv` script:

  Set MEM_ARGS=-Xms1024m -Xmx1024m -XX:MaxPermSize=256m
- For DB2, set `progressiveStreaming = 2 UnicaPlatformDS` datasource configuration.

Guidelines for deploying Unica Platform on WebSphere

You must follow a set of guidelines when you deploy Unica Platform on WebSphere.

Make sure that the version of WebSphere® meets the requirements that are described in the Recommended Software Environments and Minimum System Requirements document, including any necessary fix packs. Use the following guidelines when you deploy Unica Platform on WebSphere®:

- Specify the following custom property in the server:
    ◦ Name: `com.ibm.ws.webcontainer.invokefilterscompatibility`
    ◦ Value: true
- Set a custom property in WebSphere®.
- Deploy the EAR file or `unica.war` file as an enterprise application. When you deploy the EAR file or the `unica.war` file, ensure that the JDK source level of the JSP compiler is set to Java™ 17 for SDK 1.7 and 18 for SDK 1.8, and that JSP pages are precompiled according to the following information:
    ◦ In the form where you browse to and select the WAR file, select **Show me all installation options and parameters** so the **Select Installation Options** wizard runs.
    ◦ In step 1 of the **Select Installation Options** wizard, select **Precompile JavaServer Pages files**.
    ◦ In step 3 of the **Select Installation Options** wizard, ensure that the **JDK Source Level** is set to 17 for SDK 1.7 and set to 18 for SDK 1.8.

      If you deployed an EAR, set the JDK Source Level for each WAR file.
    ◦ In step 8 of the **Select Installation Options** wizard, select **UnicaPlatformDS** as the matching Target Resource.
    ◦ In step 10 of the **Select Installation Options** wizard, the context root must be set to `/unica`, all lower case.
- In the **Web Container Settings > Web Container > Session Management** section of the server, enable cookies. Specify a different session cookie name for each application that is deployed. Use one of the following procedures to specify a cookie name:

◦ Select the **Override session management** check box under **Session Management**.

If you deployed separate WAR files for your Unica products, in the WebSphere® console, in the **Applications > Enterprise Applications > [deployed_application] > Session Management > Enable Cookies > Cookie Name** section of the server, specify a unique session cookie name.

If you deployed EAR files for your Unica products, in the WebSphere® console, in the **Applications > Enterprise Applications > [deployed_application] > Module Management > [deployed_module] > Session Management > Enable Cookies > Cookie Name** section of the server, specify a unique session cookie name.

◦ If your installation must support non-ASCII characters, for example for Portuguese or for locales that require multi-byte characters, add the following arguments to **Generic JVM Arguments** at the server level:

-Dfile.encoding=UTF-8

-Dclient.encoding.override=UTF-8

In case you are deploying a non-production setup, add

`-DENABLE_NON_PROD_MODE=true`

In case of a production setup, this Java option must be removed or set to `false`.

Navigation tip: Select **Servers > Application Servers > Java and Process Management > Process Definition > Java Virtual Machine > Generic JVM Arguments**. See the WebSphere® documentation for additional details.

• In the **Applications > Enterprise Applications** section of the server, select the EAR file or WAR file that you deployed, then select **Class loading and update detection** and specify the following properties.

   ◦ If you are deploying a WAR file:

      ▪ For **Class loader order**, select **Classes loaded with local class loader first (parent last)**.

      ▪ For **WAR class loader policy**, select **Single class loader for application**.

   ◦ If you are deploying an EAR file:

- For **Class loader order**, select **Classes loaded with local class loader first (parent last)**.
- For **WAR class loader policy**, select **Class loader for each WAR file in application**.

- Start your deployment. If your instance of WebSphere® is configured to use a JVM version 1.7 or newer, complete the following steps to work around an issue with the time zone database.

  1. Stop WebSphere®.
  2. Download the Time Zone Update Utility for Java™ (JTZU).
  3. Follow the steps provided by the IBM® (JTZU) to update the time zone data in your JVM.
  4. Restart WebSphere®.

- In WebSphere Enterprise Applications, select *Your Application* > **Manage Modules >** *Your Application* > **Class Loader Order > Classes loaded with local class loader first (parent last)**.

- The recommended minimum heap size for the basic functioning of the application is 512 and the recommended maximum heap size is 1024.

  Complete the following tasks to specify the heap size:

  1. In WebSphere® Enterprise Applications, select **Servers > WebSphere application servers > server1 > Server Infrastructure > Java and Process Management > Process definition > Java Virtual Machine**.
  2. Set the initial heap size to 512.
  3. Set the maximum heap size to 1024

  See the WebSphere® documentation for more information about sizing.

  For DB2, set `progressiveStreaming = 2` in WebSphere console at following path: **JDBC >Data sources > UnicaPlatformDS > Custom properties**.

Guidelines for deploying Unica Platform on JBoss

You must follow a set of guidelines when you deploy Unica Platform on JBoss.

Make sure that the version of JBoss meets the requirements that are described in the HCL Enterprise Products Recommended Software Environments and Minimum System

Requirements document. Use the following guidelines when you deploy Unica Platform on JBoss:

Use the following guidelines when you deploy the Unica Platform products on any supported version of JBoss:

1. Deploy the HCL EAR file or the unica.war file as an enterprise application.

   For example: deploy <Platform_Install>\unica.war

   See https://docs.jboss.org/jbossweb/3.0.x/deployer-howto.html for instructions on Deploying Web Server Application in JBoss.

2. Complete the following tasks if your installation must support non-ASCII characters, for example for Portuguese or for locales that require multi-byte characters:

   a. Edit the `standalone.conf` script in the bin directory under your JBOSS /bin directory to add

      `-Dfile.encoding=UTF-8`

      `-Dclient.encoding.override=UTF-8`

      `-Djboss.as.management.blocking.timeout=3600`

      to `JAVA_VENDOR`.

      In case you are deploying a non-production setup, add

      `-DENABLE_NON_PROD_MODE=true`

      In case of a production setup, this Java option must be removed or set to `false`.

   b. Restart JBoss server.

3. To ensure that the Scheduler works correctly, complete the following substeps.

   • Take a backup of <JBOSS_HOME>/standalone/configuration/standalone.xml file.

   • In <JBOSS_HOME>/standalone/configuration/standalone.xml, search the module name for driver.

```
<driver name="oracledriver" module="oracle.jdbc">

<xa-datasource-class>oracle.jdbc.OracleDriver</xa-datasource-clas
s>

</driver>
```

- Add the following statement underneath `<subsystem xmlns="urn:jboss:domain:ee:4.0">` to make the module name global.

```
<global-modules>

<module name="oracle.jdbc"/>

</global-modules>
```

- Add `max-parameters="10000"` property for the following tags.

  ◦ `<http-listener name="default"...max-parameters="10000"/>`

  ◦ `<https-listener name="https...max-parameters="10000"/`

- Restart JBOSS server.

Guidelines for deploying Unica Platform on Apache Tomcat®

You must follow a set of guidelines when you deploy Unica Platform on Apache Tomcat. For Contact Central, the steps to deploy Tomcat remain same.

Make sure that the version of Apache Tomcat meets the requirements that are described in the Recommended Software Environments and Minimum System Requirements document. Use the following guidelines when you deploy Unica Platform on Apache Tomcat:

1. Deploy the HCL unica.war file as an enterprise application on Tomcat Apache server.

   Note: Deployment of HCL EAR file is not supported in Tomcat.

a. Edit the setenv.sh file for the respective product instances script in the bin directory under your Tomcat instances directory to add the following Java options:

```
- Dfile.encoding=UTF-8
```

```
-Dclient.encoding.override=UTF-8
```

In case you are deploying a non-production setup, add

```
-DENABLE_NON_PROD_MODE=true
```

In case of a production setup, this Java option must be removed or set to false.

b. Restart Tomcat.

2. If deploying in a production environment, you can add JVM heap setting for that tomcat instance in app-one/bin/setenv.sh file respectively for all the instances.

For DB2, set `progressiveStreaming = 2` in the UnicaPlatformDSdatasource configuration.

3. When Unica war is deployed on Tomcat, complete the following configuration in server.xml, where unica.war is deployed.
Add relaxedQueryChars under unica server.xml as mentioned below.

```
<Connector port="8080" protocol="HTTP/1.1"
connectionTimeout="20000"
redirectPort="8443"
relaxedQueryChars="|,[,],\,`,


{,}
,^" />
**
```

Verifying your Unica Platform installation

After you install and deploy Unica Platform, you must verify that there are no errors in your Unica Platform installation and deployment. You can configure your Unica Platform installation after the verification.

Complete the following tasks to verify your Unica Platform installation:

1. Access the Unica URL in a supported web browser.

   If you entered a domain when you installed Unica Platform, the URL is the following, where `host` is the machine where Unica Platform is installed, `domain.com` is the domain in which the host machine resides, and `port` is the port number on which the web application server listens.

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

2. Log in using the default administrator login, which is `asm_admin` with `password` as the password.

   You will be asked to change the password. You can enter the existing password, but for good security you should choose a new one.

   The default home page is the dashboard, which you will configure later.

3. Under the **Settings** menu, check the Users, User Groups, and User Permissions pages to verify that the pre-configured users, groups, roles, and permissions are present, as described in the *Unica Platform Administrator's Guide*.

4. Add a new user and group and verify that data is entered into the Unica Platform system table database.

5. Under the **Settings** menu, check the Configuration page to verify that the Unica Platform configuration properties exist.

There are additional configuration tasks, such as configuring the dashboard, setting up user access to applications, and integrating with an LDAP or web access control system (optional). See the Unica Platform Administrator's Guide for instructions.

Chapter 19. Configuring Unica Platform after deployment

For a basic installation of Unica Platform, you must configure Unica Platform after deployment if you use the Unica reporting feature, or if you want to use a password policy.

If you use the Unica reporting feature, see the Unica Reports Installation and Configuration Guide. If you want to use a password policy in mind, see Default password settings (on page 178) to determine whether you must change the default password settings.

The Unica Platform has additional properties on the **Configuration** page that perform important functions that you can optionally adjust. See the context help for the properties, or the Unica Platform Administrator's Guide to learn more about what they do and how to set them.

Additional configuration required for SSL environments

From the 12.0 release onwards, SSL for C++ components in Unica products is supported by OpenSSL.

If you have implemented SSL for C++ components (such as the Unica Campaign listener) in the 10.0.x to 11.1.x version of Unica products, you must do the following for these components.

- Create an openssl key file (.pem).
- Create new certificates using GSKit.
- Create new openssl certification key.
- Optionally create or get new certificate.
- Import openssl certification or key in certificate.
- Configure your environment to use the new certificates.

For details, see the Unica Platform Administrator's Guide , Chapter 14: Implementation of one-way SSL.

Default password settings

Unica provides default settings for using passwords. However, you can use the **General > Password settings** category on the Unica **Configuration** page to change the default settings, and create your password policies.

The default password settings apply to passwords for users that are created within Unica. The settings do not apply to the users that are imported through synchronization with an external system, such as Windows™ Active Directory, a supported LDAP directory server, or web access control server. The exception is the `Maximum failed login attempts allowed` setting, which affects both internal and external users. Also note that this property does not override any similar restriction set in an external system.

The following settings are the default password settings that Unica:

- `Maximum failed login attempts allowed` - 3
- `Password history count` - 0
- `Validity (in days)` - 30
- `Blank passwords allowed` - True
- `Allow identical user name and password` - True
- `Minimum number of numeric characters` - 0
- `Minimum number of letter characters` - 0
- `Minimum character length` - 4

See the online help for descriptions of the default settings.

Chapter 20. 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 190)* - generates a key for a client application like Unica Journey to authenticate with one Unica Platform instance.
- alertConfigTool *(on page 183)* - registers alerts and configurations for Unica products
- configTool *(on page 184)* - imports, exports, and deletes configuration settings, including product registrations
- datafilteringScriptTool *(on page 190)* - creates data filters
- encryptPasswords *(on page 192)* - encrypts and stores passwords
- encryptTomcatDBPasswords *(on page 194)* - encrypt the database passwords that the Tomcat Application server uses internally
- partitionTool *(on page 195)* - creates database entries for partitions
- populateDb *(on page 198)* - populates the Unica Platform database
- quartzjobtool *(on page 204)* - Update scheduler jobs created in version 11.1 and older versions
- restoreAccess *(on page 199)* - restores a user with the platformAdminRole role
- scheduler_console_client *(on page 202)* - 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 29. 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 *partitionName*`AdminUser`.

The partition name is automatically set as this user's password.

Examples

- Create a partition with the following characteristics.
    - Cloned from partition1
    - Partition name is `myPartition`
    - Uses the default user name (`myPartitionAdminUser`) and password (`myPartition`)
    - Uses the default group name (`myPartitionAdminGroup`)

- ◦ Description is "ClonedFromPartition1"
- ◦ `partitionTool -c -s partition1 -n myPartition -d "ClonedFromPartition1"`
- Create a partition with the following characteristics.
  - ◦ Cloned from partition1
  - ◦ Partition name is `partition2`
  - ◦ Specifies user name of `customerA` with the automatically assigned password of `partition2`
  - ◦ Specifies group name of `customerAGroup`
  - ◦ Description is "PartitionForCustomerAGroup"
  - ◦ `partitionTool -c -s partition1 -n partition2 -u customerA -g customerAGroup -d "PartitionForCustomerAGroup"`
- Update a partition with the following characteristics.
  - ◦ Cloned from partition1
  - ◦ Partition name is partition2
  - ◦ Specifiy 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 *loginNname*

Create a user with PlatformAdminRole privileges with the specified login name. Must be used with the `-p` option.

-p *password*

Specify the password for the user being created. Required with `-u`.

Examples

- Create a user with PlatformAdminRole privileges. The login name is `tempUser` and the password is `tempPassword`.

  ```
  restoreAccess -u tempUser -p tempPassword
  ```
- Change the value of the login method to `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

Chapter 21. 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.

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 30. Scripts for creating system tables

Datasource Type	Script Names
IBM® DB2®	`ManagerSchema_DB2.sql` If you plan to support multi-byte characters (for example, Chinese, Japanese, or Korean), use the `ManagerSchema_DB2_unicode.sql` script. `ManagerSchema__DB2_CeateFKConstraints.sql` `active_portlets.sql` `notification_rules.sql`
Microsoft™ SQL Server	`ManagerSchema_SqlServer.sql` `ManagerSchema__SqlServer_CeateFKConstraints.sql` `active_portlets.sql` `notification_rules.sql`
MariaDB	`ManagerSchema_MariaDB.sql` `ManagerSchema_MariaDB_StoredProcedures.sql` `ManagerSchema_MariaDB_CreateFKConstraints.sql` `active_portlets.sql` `notification_rules.sql`
Oracle	`ManagerSchema_Oracle.sql` `ManagerSchema__Oracle_CeateFKConstraints.sql` `active_portlets.sql` `notification_rules_Oracle.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 31. Scripts for enabling the Unica Scheduler

Data Source Type	Script Name
DB2®	`quartz_db2.sql`

Table 31. 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`

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.

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.

Chapter 22. Uninstalling Unica Platform

Run the Unica Platform uninstaller to uninstall Unica Platform. When you run the uninstaller, the files that were created during the installation process are removed. For example, files such as configuration files, installer registry information, and user data are removed from the computer.

When you install Unica products, an uninstaller is included in the `Uninstall_Product` directory, where Product is the name of your product. On Windows™, an entry is also added to the **Add or Remove Programs** list in the Control Panel.

If you manually remove the files in your installation directory instead of running the uninstaller, the result might be an incomplete installation if you later reinstall an product in the same location. After uninstalling a product, its database is not removed. The uninstaller only removes default files that are created during installation. Any file that is created or generated after installation is not removed.

> **Note:** On UNIX™, the same user account that installed Unica Campaign must run the uninstaller.

1. If you have deployed the Unica Platform web application, undeploy the web application from WebSphere® or WebLogic.
2. Shut down WebSphere or WebLogic.
3. Stop the processes that are related to Unica Platform.
4. If the `ddl` directory exists in the product installation directory, run the scripts that are provided in the `ddl` directory to drop tables from the system table database.
5. Make sure that the IBM SPSS® Modeler Collaboration and Deployment Service (C&DS) is uninstalled.

   Open the Installation Manager and verify that C&DS is uninstalled. If C&DS is not uninstalled, use the Installation Manager to uninstall C&DS.
6. Complete one of the following steps to uninstall Unica Platform:

- Click the Unica Platform uninstaller that exists in the `Uninstall_Product` directory. The uninstaller runs in the mode in which you installed Unica Platform.
- In a command-line window, navigate to the directory where the uninstaller exists, and run the following command to uninstall Unica Platform by using the console mode:

  Uninstall_*Product* -i console
- If you are uninstalling on a UNIX™ environment, navigate to the directory where the uninstaller exists, and run the following command to uninstall Unica Platform

  ./Uninstall\ Interaction\ History.

  📝 **Note:** UNIX™ requires spaces after \.

- In a command-line window, navigate to the directory where the uninstaller exists, and run the following command to uninstall Unica Platform by using the silent mode:

  Uninstall_*Product* -i silent

  When you uninstall Unica Platform by using the silent mode, the uninstallation process does not present any dialogs for user interaction.

📝 **Note:** If you do not specify an option for uninstalling Unica Platform, the uninstaller runs in the mode in which Unica Platform is installed.