Unica Interact V12.1.5 Upgrade Guide

Contents

Chapter 1. Upgrade overview

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

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

Upgrade roadmap

Use the upgrade roadmap to quickly find the information that you need for upgrading Unica Interact.

You can use the following table to scan the tasks that must be completed for upgrading Unica Interact:

Table 1. Unica Interact upgrade roadmap

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

Topic	Information
Upgrade overview *(on page 1)*	This chapter provides the following information: • How the installers work *(on page 3)* • Modes of installation *(on page 4)* • Unica Interact documentation and help *(on page 6)*
Planning the Unica Interact upgrade *(on page 10)*	This chapter provides the following information: • Prerequisites *(on page)* • Upgrade prerequisites for all Unica products *(on page 14)* • Unica Interact upgrade tools *(on page 16)*

Table 1. Unica Interact upgrade roadmap

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

Topic	Information
	• Unica Interact upgrade worksheet *(on page 23)* • Information for creating JDBC connections *(on page 29)*
Upgrading Unica Interact *(on page 42)*	This chapter provides the following information: • Backing up the Unica Interact runtime environment *(on page 42)* • Undeploying the Unica Interact runtime server *(on page 43)* • Running the installer *(on page 43)* • Reviewing and modifying the SQL upgrade script *(on page 44)* • Setting environment variables *(on page 49)* • Running the Unica Interact upgrade tools *(on page 54)* • Redeploying the Unica Interact runtime server in the web application server *(on page 57)* • Upgrade log *(on page 57)* • Upgrading partitions *(on page 58)* • Creating and populating the Unica Interact system tables *(on page 58)*
Deploying Unica Interact *(on page 64)*	This chapter provides information: on deploying Unica Interact.
Uninstalling Unica Interact *(on page 78)*	This chapter provides information about how to uninstall Unica Interact.

Table 1. Unica Interact upgrade roadmap

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

Topic	Information
configTool *(on page 81)*	This chapter provides information about how to use the configTool utility.

How the installers work

You must use the suite installer and the product installer when you install or upgrade any Unica product. For example, for installing Unica Plan, you must use the Unica suite installer and the Unica Plan installer.

Make sure that you use the following guidelines before you use the Unica suite installer and the product installer:

- The Unica installer and the product installer must be in the same directory on the computer where you want to install the product. When multiple versions of a product installer are present in the directory with the Unica installer, the Unica installer always shows the latest version of the product on the Unica Products screen in the installation wizard.
- If you are planning to install a patch immediately after you install an Unica product, make sure that the patch installer is in the same directory as that of the suite and product installers.
- The default top-level directory for Unica installations is `/HCL/Unica` for UNIX™ or `C:\HCL\Unica` for Windows™. However, you can change the directory during installation.

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 Interact .

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

GUIX Window System mode mode

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

UNIX™ X Window System mode

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

Console mode

Use the console mode to install Unica Interact 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 Interact 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 Interact. 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 Interact 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 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.properties` 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*`.properties`	The sample response file for the Unica Interact installer. For example, `installer_uc`*n.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_ump`*n.n.n.n*`.properties` is the response file of the Unica Platform installer, where n.n.n.n is the version number. For example, `installer_ul`*n.n.n.n*`.properties` is the response file of the Leads installer, where n.n.n.n is the version number.

Unica Interact documentation and help

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

Use the following table to get information about how to get started with Unica Interact:

Table 4. Get up and running

This two-columned table provides information about the tasks in one column, and documentation in the second column.

Task	Documentation
View a list of new features, known issues, and workarounds	*Unica Interact Release Notes®*

Table 4. Get up and running

This two-columned table provides information about the tasks in one column, and documentation in the second column.

(continued)

Task	Documentation
Learn about the structure of the Unica Interact database	*Unica Interact System Tables and Data Dictionary*
Install or upgrade Unica Interact and deploy the Unica Interact web application	One of the following guides: • *Unica Interact Installation Guide* • *Unica Interact Upgrade Guide*
Implement the IBM Cognos® reports provided with Unica Interact	Cognos Reports Installation and Configuration Guide
Implement the Unica Insights reports provided with Unica Interact	Unica Insights Reports Installation and Configuration Guide

Use the following table to get information about how to configure and use Unica Interact:

Table 5. Configure and use Unica Interact

This two-columned table provides information about the tasks in one column, and documentation in the second column.

Task	Documentation
• Maintain users and roles • Maintain data sources • Configure Unica Interact optional offer serving features • Monitor and maintain runtime environment performance	*Unica Interact Administrator's Guide*

Table 5. Configure and use Unica Interact

This two-columned table provides information about the tasks in one column, and documentation in the second column.

(continued)

Task	Documentation
• Work with interactive channels, events, learning models, and offers • Create and deploy interactive flowcharts • View Unica Interact reports	Unica Interact User's Guide
Use Unica Interact macros	Macros for Unica: User's Guide
Adjust components to obtain optimal performance	Unica Interact Tuning Guide

Use the following table to get information about how to get help if you face issues when you use Unica Interact:

Table 6. Get help

This two-columned table provides information about the tasks in one column, and documentation in the second column.

Task	Instructions
Open online help	1. Choose **Help > Help for this page** to open a context-sensitive help topic. 2. Click the **Show Navigation** icon in the help window to display the full help.
Obtain PDFs	Use either of the following methods:

Table 6. Get help

This two-columned table provides information about the tasks in one column, and documentation in the second column.

(continued)

Task	Instructions
	• Choose **Help > Product Documentation** to access Unica Interact PDFs. • Choose **Help > All HCL Unica Documentation** to access all available documentation.

Chapter 2. Planning the Unica Interact upgrade

Upgrade your installation of Unica Interact after understanding the guidelines that are specific to your current version of Unica Interact.

Unica Interact supports the following upgrade paths:

- 12.1.x → 12.1.5
- 12.1.0.x → 12.1.5

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 12.1.5.

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 12.1.5.

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 12.1.5.
- **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 12.1.5.

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.

All HCL Unica products must be at the same release level.

Unica Interact version: This version requires Unica Interact base version 12.1, 12.1.0.1,12.1.0.2, 12.1.0.3, 12.1.0.4, 12.1.1, 12.1.2, 12.1.3, or 12.1.4. The following are other HCL Unica product version dependencies.

- Unica Platform must be at version 12.1.5.
- Unica Campaign must be at version 12.1.5.

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.

Important: For best performance, install Campaign listener to execute Optimize session on its own system, where no other Unica products are installed. Unica Optimize requires significant computation and data processing resources. You have the greatest control and flexibility for performance-tuning if you operate Unica Optimize in a dedicated environment.

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.

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.

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.

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.

> **Note:** For versions 12.0.0 and later, to execute Optimize sessions, users are required to apply for licenses. For more details, contact the HCL Support or Sales team.

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:** For installations on UNIX, you may require to set the `Djava.awt.headless` property to true in your web application server. The setting is required only when you are unable to view Unica Optimize reports. See the Unica Campaign Installation Guide for details. You do not require to prepare any additional data sources for Unica Optimize because Unica Optimize uses the Unica Campaign system tables data source.

> **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.

JDK requirements

To integrate Unica Interact with MQ, Unica Interact runtime must be on appserver with JDK 1.8. For WebSphere and WebLogic, it is recommended to use the latest supplied JDK fix pack version.

Upgrade prerequisites for all Unica products

Meet all requirements for permissions, operating system, and knowledge correctly before you upgrade Interact 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 Interact 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 Interact, 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 12.1.5, 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 12.1.5 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 12.1.5 installs IBM JRE.

Unica Interact upgrade tools

You must upgrade the runtime environment and the design time environment when you upgrade Unica Interact. Run the Unica Interact upgrade tools to upgrade system tables, contact and response history tables, and Unica Interact user profile tables.

Unica Interact provides five upgrade tools, one for upgrading the design time environment (aciUpgradeTool) and four for upgrading the runtime environment (aciUpgradeTool_crhtab, aciUpgradeTool_lrntab, aciUpgradeTool_runtab, and aciUpgradeTool_usrtab). The upgrade scripts are delivered with the new version of Unica Interact, and are available after you run the Unica Unica Interact installer in clean or upgrade mode for both the runtime environment and the design time environment.

You can upgrade the Unica Interact design time environment configuration properties when you upgrade the Unica Campaign configuration properties.

Use the following table to understand the purpose of the Unica Interact upgrade tools:

Table 7. Unica Interact upgrade tools

This three-columned table describes the Unica Interact upgrade tools in the first column, the location of the tools in the second column, and the purpose of the tools in the third column.

Tool	Location	Purpose
aciUpgradeTool	*Interact_Design_Install_Directory* `/interactDT/tools/upgrade`	Upgrades the Unica Interact design time environment tables in the Unica

Table 7. Unica Interact upgrade tools

This three-columned table describes the Unica Interact upgrade tools in the first column, the location of the tools in the second column, and the purpose of the tools in the third column.

(continued)

Tool	Location	Purpose
		Campaign system tables.
aciUpgradeTool_runtab	*Interact_Runtime_Install_Directory* /tools/upgrade	Upgrades the Unica Interact runtime environment tables and the Unica Interact runtime environment configuration properties.
aciUpgradeTool_lrntab	*Interact_Runtime_Install_Directory*/tools/upgrade	Upgrades the Unica Interact learning tables.
aciUpgradeTool_crhtab	*Interact_Runtime_Install_Directory*/tools/upgrade	Upgrades the contact and response history tables that are used with cross-session response tracking.
aciUpgradeTool_usrtab	*Interact_Runtime_Install_Directory*/tools/upgrade	Upgrades the Unica Interact tables that are required in your profile user tables.

Strategy migration utility tool

A utility is provided to migrate the strategies created using the old Strategy user interface prior to version 12.0 to the new models so that they can be viewed and edited on the new Strategy user interface. In addition, this strategy migration tool can be used to revert the migration, i.e., smart strategies introduced in Unica Interact version 12.0 to the older model.

> **Note:**
>
> This migration utility only needs to be run when upgrading Interact from version prior to 12.0 to any version up to 12.1.0.2. From 12.1.0,3 onwards, Interact DT Upgrade Tool will run this utility automatically to migrate all old strategies to smart strategies. For usage, see the Interact Upgrade Guides of previous versions.

Migration of triggered action

The TriggeredActionDBUpgradeTask migrates the data from old triggered action to new triggered action. This task runs along with existing upgrade utility tool.

The new triggered action contents are saved into UACI_EventAction , UACI_TriggeredAction, UACI_TABranch, and UACI_TAOffer tables.

During migration, old triggered actions data are retrieved from the Details column in UACI_EventAction table and saved in UACI_TriggeredAction, UACI_TABranch, and UACI_TAOffer tables.

> **Note:**

1. Only Old triggered action with one value (Enable) of UACI_EventAction.ActionValue column is migrated

2. The migration of already migrated triggered action is skipped.

3. Migrated triggered action name is "MigratedTA_<EventActionID>" for column UACI_TriggeredAction.name

4. 'Week' channel delay type should be converted into 'day' delay type as 'week' delay type in new triggered action is not supported.

5. If there are multiple channels for a branch (eligibility/offer page data) in old triggered action, then multiple branches for each channel are created, i.e., creating multiple branches with same eligibility/offer data by differing only channel details.

6. In case of multiple offers, an offerlist is created on fly which have all these offers and this offerlist is created under InteractTA folder.

7. The following attributes were not used in old triggered action, so it loads in UI with default values: scorePredicateEnabled, scorePredicate, eligibilityPredicateEnabled, eligibilityPredicate, effectiveDate, expirationDate, learningMode, lmName, learningModelId, offerAttrExpressions

8. Since, the table name is not recieved along with the attribute name, so it is checked that particular attribute in all audience level, whereever it matches that option is selected.

In case, if that attributes belongs to multiple audience, then the first matched attribute is set in any audience level. Once, we initiate saving of migrated TA in new UI, it i saved with table name.

Migration of gateway groups

1. For fresh installation, the outbound channel (gateway groups) is hid from Platform configuration and new users cannot add new outbound channel (gateway groups) from configuration as the template to add a new outbound channel is removed.
2. For migrated setup, if configuration is not hidden for outbound channel (gateway groups):
    - Migrated outbound channels (gateway groups) is read-only.
    - The outbound channel template is removed so the user cannot add new outbound channel (gateway groups) from configuration.

ILPB tables upgrade utility tool

The Strategy user interface is redesigned for better usability. You can use score predicate and eligibility predicate simultaneously. The ILPB migration upgrade utility migrates the existing Interact List process box (ILPB) tables to create new fields and populate with values from existing predicate fields.

- The existing structure of the ILPB tables remain valid. The users can decide not to use this utility, if they do not want new fields for existing ILPB tables.
- The users can migrate the existing ILPB table and use the migrated table to create a new ILPB through UI.

Usage

This utility facilitates the migration of data from old predicate and enableStateId fields to new predicate fields scorepredicate, scorepredicateenabled, eligibilitypredicate and eligibilitypredicateenabled for the ILPB tables and add effectivedate and expirationdate for record eligibility.

- **Existing fields**

  The following are the existing fields in ILPB tables, which facilitate the user to map predicate field as score predicate or eligibility predicate

  - Predicate
  - EnableStateId

  If the value of EnableStateId field in ILPB is mapped to 3, the expression in predicate field is used as score predicated. If the value of EnableStateId field is mapped to 2, the expression in predicate field is used as eligibility predicated.

- **New fields**

  The following are the new fields added by the utility.

Name	Data Type	Description
ScorePredica-teEnabled	Nu-meric	Valid values are 0 or 1. Any numeric value other than 1 is treated as 0.

ScorePredicate	Text	When ScorePredicateEnabled is set to 1, the column value is used as score predicate.
EligibilityPredicateEnabled	Numeric	Valid values are 0 or 1. Any numeric value other than 1 is treated as 0.
EligibilityPredicate	Text	When EligibilityPredicateEnabled is set to 1, the column value is used as eligibility predicate.
EffectiveDate	Timestamp	The date from when the ILPB record is effective. Null is considered as a record valid.
ExpirationDate	Timestamp	The date when the ILPB record must expire. Null is considered as the record not expired.

• **Migration rules**

  ◦ If the Predicate and EnableStateId fields are not present in the table, the table is ignored by the migration utility.

  ◦ New fields ScorePredicateEnabled, ScorePredicate, EligibilityPredicateEnabled, EligibilityPredicate, EffectiveDate and ExpirationDate is added to the ILPB table, if not already present.

  ◦ The values of the EnableStateId and Predicate columns are migrated on following conditions.

    ▪

Value of column EnableStateID	Migrated column values
2	EligibilityPredicateEnabled = 1 EligibilityPredicate = Predicate ScorePredicateEnabled = 0 ScorePredicate = null
3	EligibilityPredicateEnabled = 0

	EligibilityPredicate = null
	ScorePredicateEnabled = 1
	ScorePredicate = Predicate
1	EligibilityPredicateEnabled = 0
	EligibilityPredicate = null
	ScorePredicateEnabled = 0
	ScorePredicate = null

- ◦ EffectiveDate and ExpirationDate is populated with initial values as null. Note that null is considered as the valid dates for the ILPB record.
- ◦ The Utility can run for multiple times for the same table. If the utility is run for an already migrated table, the values of the new predicate fields is updated as per the latest values in predicate or enableStateID fields. This refreshes the latest values in the new predicate fields since the new predicate fields for existing ILPBs through UI are not available.
- ◦ Predicate and EnableStateId fields are dropped from the table after migration.

Properties setups for the ILPB migration upgrade utility

Users must navigate to the path `<Installation_Directory>\Interact\tools\upgrade\conf` and open **ACIILPBUpgradeTaskList_usrtab.properties** file for doing the properties setup.

Users must change the following properties:

- • **ILPB_TABLES_TO_UPDATE** – This appends the ILPB (whitelist offers, default offers, offer by SQL) table names which you want to migrate data from old fields to new predicate fields. The utility will only work on the tables mentioned for this property. The utility will create new predicate fields, if not already present, populate new

predicate fields based on the existing predicate column values and drop these old columns.

- **ILPB_MIGRATIONTASK_BATCHSIZE** - Users can set this property for specifying the batch size for data update operations of this utility. The default value is 5000. It indicates the number of record processed at a time for update operation

Procedure for running the ILPB migration upgrade

Users must run the standalone batch file `aciILPBUpgradeTool_usrtab.bat` or the shell script file `aciILPBUpgradeTool_usrtab.sh` from path `<Installation_Directory>\Interact\tools\upgrade` to run the migration utility.

Unica Interact upgrade worksheet

Use the Unica Interact upgrade worksheet to gather information about the database that contains your Unica Interact upgrade system tables and about other Unica products that are required for upgrading Unica Interact.

Unica Platform database information

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

Information required to upgrade the Unica Interact runtime environment

Gather information about your Unica Interact runtime installation before you run the Unica Interact runtime environment upgrade tools.

aciUpgradeTool_runtab

Collect the following information about the configuration of the target system:

- The directory where Unica Platform is installed.
- Full path of the Unica Interact configuration file (`interact_configuration.xml`). The file is in the `conf` directory under the Unica Interact installation.

If you connect to the runtime environment system tables by using the web application server, collect the following information:

- Host name
- Port
- User name
- Password
- For WebLogic: Full path and file name of the WebLogic JAR file

If you connect to the runtime environment system tables by using JDBC, collect the following information:

- Java™ class name for the JDBC driver
- JDBC URL
- Additional properties that are required by the JDBC driver
- Database user name and password

Collect the following information about the target runtime environment database:

- Catalog (or database) containing the target runtime environment system tables
- Schema
- Whether the tables are configured for Unicode

Consider the following points.

- If Learning is enabled, for version 2, using SampleMethod1, for version 1, using SampleMethod1.
- During the interact configuration process, system verifies if the setting are matching.
- During the interact offer treatment optimizer process in learning version2, the offer learning is handles based on SampleMethod1 and SampleMethod2 setting.
- The offer RWA is calculated based on SampleMethod1 and SampleMethod2 settings.

Collect the following information about the Unica Interact installation on the source system:

- Version of Unica Interact you are upgrading from

aciUpgradeTool_lrntab

Collect the following information about the configuration of the target system:

- The directory where Unica Platform is installed

If you connect to the learning tables by using the web application server, collect the following information:

- Host name
- Port
- User name
- Password
- For WebLogic: Full path and file name of the WebLogic JAR file

If you connect to the learning tables by using JDBC, collect the following information:

- Java™ class name for the JDBC driver
- JDBC URL
- Additional properties that are required by the JDBC driver
- Database user name and password

Collect the following information about the target learning database:

- Catalog (or database) containing the target learning tables
- Schema
- Whether the tables are configured for Unicode

Collect the following information about the Unica Interact installation on the source system:

- Version of Unica Interact you are upgrading from

aciUpgradeTool_crhtab

Collect the following information about the configuration of the target system:

- The directory where Unica Platform is installed

If you connect to the contact history tables for cross-session response by using the web application server, collect the following information:

- Host name
- Port
- User name
- Password
- For WebLogic: Full path and file name of the WebLogic JAR file

If you connect to the contact history tables for cross-session response by using JDBC, collect the following information:

- Java™ class name for the JDBC driver
- JDBC URL
- Additional properties that are required by the JDBC driver
- Database user name and password

Collect the following information about the target contact history tables for cross-session response database:

- Catalog (or database) containing the target contact history tables for cross-session response
- Schema
- Whether the tables are configured for Unicode

Collect the following information about the Unica Interact installation on the source system:

- Version of Unica Interact you are upgrading from

aciUpgradeTool_usrtab

Collect the following information about the configuration of the target system:

- The directory where Unica Platform is installed

If you connect to the user profile tables by using the web application server, collect the following information:

- Host name
- Port
- User name
- Password
- For WebLogic: Full path and file name of the WebLogic JAR file

If you connect to the user profile tables by using JDBC, collect the following information:

- Java™ class name for the JDBC driver
- JDBC URL
- Additional properties that are required by the JDBC driver
- Database user name and password

Collect the following information about the target user profile database:

- Catalog (or database) containing the target user profile tables
- Schema
- Whether the tables are configured for Unicode

Collect the following information about the Unica Interact installation on the source system:

- Version of Unica Interact you are upgrading from

Information required to upgrade the Unica Interact design time environment

Gather information about your Unica Interact design time installation before you run the Unica Interact design time environment upgrade tool.

aciUpgradeTool

Collect the following information about the configuration of the target system:

- The name of the partition you are upgrading.
- The directory where Unica Platform is installed.
- Full path to the Unica Campaign configuration file (`campaign_configuration.xml`). The Unica Campaign configuration file is in the `conf` directory under your Unica Campaign installation.

If you connect to the design time environment system tables by using the web application server, collect the following information:

- Host name
- Port
- User name
- Password
- For WebLogic: Full path and file name of the WebLogic JAR file

If you connect to the design time environment system tables by using JDBC, collect the following information:

- Java™ class name for the JDBC driver
- JDBC URL
- Additional properties that are required by the JDBC driver
- Database user name and password

Collect the following information about the target design time environment database:

- Catalog (or database) containing the target design time environment system tables
- Schema
- Whether the tables are configured for Unicode

Collect the following information about the Unica Interact installation on the source system:

- Version of Unica Interact that you are upgrading from

Information for creating JDBC connections

Use default values when you create JDBC connections if specific values are not provided. For more information, see the application server documentation.

> ✎ **Note:** If you are not using the default port setting for your database, make sure that you change it to the correct value.

WebLogic

Use the following values if your application server is WebLogic:

SQLServer

- Database Driver: Microsoft™ MS SQL Server Driver (Type 4) Versions: 2012, 2012 SP1 and SP3, 2014, 2014 SP1, 2016 SP1
- Default port: 1433
- Driver class: `com.microsoft.sqlserver.jdbc.SQLServerDriver`

- Driver URL: `jdbc:sqlserver://<your_db_host>[\` `\<named_instance>]:<your_db_port>;databaseName=<your_db_name>`
- Properties: Add `user=<your_db_user_name>`

Oracle

- Driver: Other
- Default port: 1521
- Driver class: `oracle.jdbc.OracleDriver`
- Driver URL:

  `jdbc:oracle:thin:@<your_db_host>:<your_db_port>:<your_db_service_name>`

  Enter the driver URL by using the format that is shown. Unica applications do not allow the use of Oracle's RAC (Real Application Cluster) format for JDBC connections.
- Properties: Add `user=<your_db_user_name>`

DB2®

- Driver: Other
- Default port: 50000
- Driver class: `com.ibm.db2.jcc.DB2Driver`
- Driver URL: `jdbc:db2://<your_db_host>:<your_db_port>/` `<your_db_name>`
- Properties: Add `user=<your_db_user_name>`

MariaDB

- Database type: MYSQL
- Driver: mariadb-java-client-2.5.1.jar
- Default port: 3306
- Driver class: org.mariadb.jdbc.Driver
- Driver URL: jdbc:mariadb://<your_db_host>:<PORT>/<Your_DB_user_name>
- Properties: Add user=<your_db_user_name>
- Password=<your_db_user_password>

> ✎ **Note:** From version 12.1.0.2, use:
>
> - MariaDB ODBC connector version 3.1.6 for Windows and Linux
> - MariaDB ODBC connector version 3.1.9 for SUSE operating system

MariaDB ODBC connector versions

Microsoft Windows	**Maria DB ODBC Connector**: 3.1.6 **Download link**: https://downloads.mariadb.com/Connectors/odbc/connector-odbc-3.1.6/mariadb-connector-odbc-3.1.6-win64.msi
RHEL	**Maria DB ODBC Connector**: 3.1.6 **Download link**: https://downloads.mariadb.com/Connectors/odbc/connector-odbc-3.1.6/mariadb-connector-odbc-3.1.6-ga-rhel7-x86_-64.tar.gz
SUSE	**Maria DB ODBC Connector**: 3.1.9 **Download link**: https://downloads.mariadb.com/Connectors/odbc/connector-odbc-3.1.9/mariadb-connector-odbc-3.1.9-sles123-amd64.tar.gz

MariaDB ODBC driver level configurations

Example for Microsoft Windows: Set the following configuration in ODBC Driver Manager following the **Statements** parameter:

```
SET SQL_MODE='PIPES_AS_CONCAT,ANSI_QUOTES,IGNORE_SPACE,NO_KEY_OPTIONS';
```

Example for SUSE/Linux: Set the following configuration in `odbc.ini`:

```
[<CampaignSystem>]

Driver=/usr/local/mariaDBConnector/lib/libmaodbc.so

DATABASE=<DATABASE_NAME>

DESCRIPTION=MariaDB

SERVER=<DATABASE_SERVER>

UID=<USER_NAME>

PASSWORD=<PASSWORD>

PORT=3306

InitStmt=SET

SQL_MODE='PIPES_AS_CONCAT,ANSI_QUOTES,IGNORE_SPACE,NO_KEY_OPTIONS';
```

⚠️ **Attention:** Use MariaDB database user name. Do not use `root` and `unica*03` as username and password for `UA_SYSTEM_TABLES` datasource.

MariaDB JAVA Client (JDBC Client): Download the version 2.4.0 client using the following link: https://mariadb.com/kb/en/library/about-mariadb-connector-j/

📝 **Note:** For AIX, we do not support MariaDB as System and User Database.

WebSphere®

Use the following values if your application server is WebSphere®:

SQLServer

- Driver: N/A
- Default port: 1433

- Driver class: `com.microsoft.sqlserver.jdbc.SQLServerConnectionPoolDataSource`
- Driver URL: jdbc:sqlserver://<DBhostName>:1433;databaseName=<DBName>

In the **Database Type** field, select **User-defined**.

After you create the JDBC Provider and data source, navigate to the **Custom Properties** for the data source, and add or modify properties as follows.

- `serverName=<your_SQL_server_name>`
- `portNumber =<SQL_Server_Port_Number>`
- `databaseName=<your_database_name>`

Add the following custom properties.

- Name: `webSphereDefaultIsolationLevel`
- Value: `1`
- Datatype: `Integer`

Oracle

- Driver: Oracle JDBC Driver
- Default port: 1521
- Driver class: `oracle.jdbc.OracleDriver`
- Driver URL:
  `jdbc:oracle:thin:@<your_db_host>:<your_db_port>:<your_db_service_name>`

  Enter the driver URL by using the format that is shown. Unica applications do not allow the use of Oracle's RAC (Real Application Cluster) format for JDBC connections.

DB2®

- Driver: JCC Driver Provider
- Default port: 50000
- Driver class: `com.ibm.db2.jcc.DB2Driver`
- Driver URL: `jdbc:db2://<your_db_host>:<your_db_port>/`
  `<your_db_name>`

To add the custom properties, complete the following steps.

1. Click the data source that you created. Go to the Custom Properties for the data source.
2. Select the **Custom properties** link.
3. Set the value for the `resultSetHoldability` property to `1`. If you do not see the `resultSetHoldability` property, create the `resultSetHoldability` property and set its value to `1`.
4. Set the value for the `webSphereDefaultIsolationLevel` property to `2`. If you do not see the `webSphereDefaultIsolationLevel` property, create the `webSphereDefaultIsolationLevel` property and set its value to `2`.

The following are the custom properties.

- Name: `webSphereDefaultIsolationLevel`
- Value: `2`
- Datatype: `Integer`

MariaDB

- Database type: User_define
- Implementation class name = org.mariadb.jdbc.MySQLDataSource
- Driver: mariadb-java-client-2.5.1.jar

  JNDI custom properties
- Component-managed authentication alias = Select the authentication details for user
- Mapping-configuration alias = WSLogin
- portNumber = 3306
- relaxAutoCommit=true
- databaseName = <DATABASE NAME>
- serverName = the host name of the MariaDB server
- user = the user name of the MariaDB server
- password = the password associated with the user name

> **Note:** From version 12.1.0.2, use:
>
>   • MariaDB ODBC connector version 3.1.6 for Windows and Linux
>   • MariaDB ODBC connector version 3.1.9 for SUSE operating system

MariaDB ODBC connector versions

Microsoft Windows	**Maria DB ODBC Connector**: 3.1.6 **Download link**: https://downloads.mariadb.com/Connectors/odbc/connector-odbc-3.1.6/mariadb-connector-odbc-3.1.6-win64.msi
RHEL	**Maria DB ODBC Connector**: 3.1.6 **Download link**: https://downloads.mariadb.com/Connectors/odbc/connector-odbc-3.1.6/mariadb-connector-odbc-3.1.6-ga-rhel7-x86_-64.tar.gz
SUSE	**Maria DB ODBC Connector**: 3.1.9 **Download link**: https://downloads.mariadb.com/Connectors/odbc/connector-odbc-3.1.9/mariadb-connector-odbc-3.1.9-sles123-amd64.tar.gz

MariaDB ODBC driver level configurations

Example for Microsoft Windows: Set the following configuration in ODBC Driver Manager following the **Statements** parameter:

```
SET SQL_MODE='PIPES_AS_CONCAT,ANSI_QUOTES,IGNORE_SPACE,NO_KEY_OPTIONS';
```

Example for SUSE/Linux: Set the following configuration in `odbc.ini`:

```
[<CampaignSystem>]

Driver=/usr/local/mariaDBConnector/lib/libmaodbc.so

DATABASE=<DATABASE_NAME>

DESCRIPTION=MariaDB

SERVER=<DATABASE_SERVER>

UID=<USER_NAME>

PASSWORD=<PASSWORD>

PORT=3306

InitStmt=SET

SQL_MODE='PIPES_AS_CONCAT,ANSI_QUOTES,IGNORE_SPACE,NO_KEY_OPTIONS';
```

> ⚠️ **Attention:** Use MariaDB database user name. Do not use `root` and `unica*03` as username and password for `UA_SYSTEM_TABLES` datasource.

MariaDB JAVA Client (JDBC Client): Download the version 2.4.0 client using the following link: https://mariadb.com/kb/en/library/about-mariadb-connector-j/

> ✏️ **Note:** For AIX, we do not support MariaDB as System and User Database.

Tomcat

Use the following values if your application server is Tomcat:

MariaDB

- Driver: mariadb-java-client-2.5.1.jar
- Default port: 3306
- Driver class: org.mariadb.jdbc.Driver

- Driver URL: jdbc:mariadb://<your_db_host>:<PORT>/<Your_DB_user_name>
- Properties: Add user=<your_db_user_name>
- Properties: Add password=<your_db_password>

JBoss

Specify the native library path of the database driver JAR file on your server.

Use the following values if your application server is JBoss:

SQL Server

- Database Driver: Microsoft MS SQL Server Driver (Type 4) Versions: 2012, 2012 SP1 and SP3, 2014, 2014 SP1, 2016 SP1
- Default port: 1433
- Driver class: com.microsoft.sqlserver.jdbc.SQLServerDriver
- Driver URL: jdbc:sqlserver:// <your_db_host>:<your_db_port>;databaseName=<your_db_name>,validconnection- checker-class>
- name: org.jboss.jca.adapters.jdbc.extensions.mssql.MSSQLValidConnectionChecker

  For example:

  ```
  /subsystem=datasources/data-source=UnicaPlatformDS:add(jndiname="
  java:/UnicaPlatformDS",connection-url="jdbc:sqlserver://
  localhost:1433;databaseName=plat11",driver-name=sql,username=
  sa,password=test1234,valid-connection-checker-
  classname="org.jboss.jca.adapters.jdbc.extensions.mssql.MSSQLValidConnectionChecker")
  **
  ```

Oracle

- Driver: Oracle JDBC Driver
- Default port: 1521
- Driver class: oracle.jdbc.OracleDriver
- Driver URL:jdbc:oracle:thin:@<your_db_host>:<your_db_port>:<your_db_service_name>

DB2

- Driver: JCC Driver Provider
- Default port: 50000
- Driver class: com.ibm.db2.jcc.DB2Driver
- Driver URL: jdbc:db2://<your_db_host>:<your_db_port>/<your_db_name>
- valid-connection-checker class-name=
  org.jboss.jca.adapters.jdbc.extensions.db2.DB2ValidConnectionChecker

MariaDB

- Database Driver: mariadb-java-client-2.5.1.jar
- Default port: `3306`
- Driver class: org.mariadb.jdbc.Driver
- Driver URL: jdbc:mariadb://<your_db_host>:<PORT>/<Your_DB_user_name>
- Properties: `Add user=<your_db_user_name>`
- Properties: `Add password=<your_db_password>`
- Driver module xa-datasource-class= org.mariadb.jdbc.MySQLDataSource

> **Note:** From version 12.1.0.2, use:
>
> - MariaDB ODBC connector version 3.1.6 for Windows and Linux
> - MariaDB ODBC connector version 3.1.9 for SUSE operating system

MariaDB ODBC connector versions

Microsoft Windows	**Maria DB ODBC Connector**: 3.1.6
	Download link:
	https://downloads.mariadb.com/Connectors/odbc/connector-odbc-3.1.6/mariadb-connector-odbc-3.1.6-win64.msi

RHEL	**Maria DB ODBC Connector**: 3.1.6
	Download link:
	https://downloads.mariadb.com/Connectors/odbc/connector-odbc-3.1.6/mariadb-connector-odbc-3.1.6-ga-rhel7-x86_-64.tar.gz
SUSE	**Maria DB ODBC Connector**: 3.1.9
	Download link:
	https://downloads.mariadb.com/Connectors/odbc/connector-odbc-3.1.9/mariadb-connector-odbc-3.1.9-sles123-amd64.tar.gz

MariaDB ODBC driver level configurations

Example for Microsoft Windows: Set the following configuration in ODBC Driver Manager following the **Statements** parameter:

```
SET SQL_MODE='PIPES_AS_CONCAT,ANSI_QUOTES,IGNORE_SPACE,NO_KEY_OPTIONS';
```

Example for SUSE/Linux: Set the following configuration in `odbc.ini`:

```
[<CampaignSystem>]

Driver=/usr/local/mariaDBConnector/lib/libmaodbc.so

DATABASE=<DATABASE_NAME>

DESCRIPTION=MariaDB

SERVER=<DATABASE_SERVER>

UID=<USER_NAME>

PASSWORD=<PASSWORD>

PORT=3306

InitStmt=SET
```

```
SQL_MODE='PIPES_AS_CONCAT,ANSI_QUOTES,IGNORE_SPACE,NO_KEY_OPTIONS';
```

⚠️ **Attention:** Use MariaDB database user name. Do not use `root` and `unica*03` as username and password for `UA_SYSTEM_TABLES` datasource.

MariaDB JAVA Client (JDBC Client): Download the version 2.4.0 client using the following link: https://mariadb.com/kb/en/library/about-mariadb-connector-j/

📝 **Note:** For AIX, we do not support MariaDB as System and User Database.

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 Unica Interact

You can upgrade Unica Interact by overwriting your existing Unica Interact installation. If you cannot upgrade your current version of Unica Interact directly, you must install Unica Interact in a new location.

An in-place upgrade is one where you overwrite your existing installation. You can complete in-place upgrades for Unica Interact version 12.1.5.

To ensure that the installer automatically upgrades your existing Unica Interact design time and runtime environment, select the same location as your old Unica Interact design time and runtime location.

When in-place upgrades are not possible, you must install Unica Interact in a new location. Because of the architectural changes between Unica Interact version 8.5.0 and previous versions of Unica Interact, there is no upgrade path from earlier versions of Unica Interact.

Complete the following steps to upgrade Unica Interact:

1. Back up the Unica Interact runtime environment.
2. Undeploy the Unica Interact runtime server.
3. Run the Unica installer.
4. Review and modify the SQL upgrade script.
5. Set environment variables.
6. Run the upgrade tool for the Unica Interact design time environment.
7. Run the upgrade tools for the Unica Interact runtime environment
8. Redeploy the Unica Interact runtime server in the web application server
9. Check the upgrade log

Backing up the Unica Interact runtime environment

Before you upgrade Unica Interact, back up all the files, system table database, and configuration settings that are used by the Unica Interact runtime environment to prevent loss of data and configuration settings.

✏️ **Note:** You must back up only one Unica Interact runtime server per server group.

If your Unica Interact runtime environment installation requires any of the configuration settings from your old Unica Interact version in addition to the new (default) settings in the new version, use the configTool utility to export the old Unica Interact configuration parameters. Specify a different file name for the `exported.xml` file and note the location where you save it.

Undeploying the Unica Interact runtime server

Before you upgrade Unica Interact, you must undeploy the Unica Interact runtime server so that the Unica Interact installer can complete a clean and error-free upgrade.

You must undeploy the Unica Interact runtime server so that the web application server releases the lock on the `InteractRT.war` file, which is updated during the Unica Interact upgrade. Releasing the lock on the `interactRT.war` file allows the Unica Interact installer to cleanly update the `interactRT.war` file and register the new version of Unica Interact in the Unica console.

Complete the following steps to undeploy the Unica Interact runtime server:

1. Follow the instructions in your web application server to undeploy the `interactRT.war` file, and save or activate all changes.
2. Shut down and restart the web application server after you undeploy the Unica Interact runtime server to ensure that the lock on the `InteractRT.war` file is released.

Running the installer

You must run the Unica installer to upgrade Unica Interact. The Unica installer starts the Unica Interact installer during the process.

After you undeploy the Unica Interact runtime environment, run the Unica installer. When the installer prompts you to select the Unica product that you want to install, select Unica

Interact. The Unica Interact installer starts. The Unica Interact installer detects that you have an earlier version installed and runs in upgrade mode.

You can install or upgrade the following Interact components:

- Unica Interact Runtime Environment
- Unica Interact Design Time Environment
- Unica Interact Pattern State ETL
- Unica Insights reports

Note: Extreme Scale is not supported in the 11.1 release. In case of upgrade to 11.1, if the base has the CacheManager as 'EHCache' with 'CacheType' as 'Distributed' or CacheManager as 'ExtremeScale', then the upgrade would update to 'CacheManager' as Ignite and 'CacheType' as 'Distributed'.

After you finish upgrading Unica Interact, you must deploy the Unica Interact runtime environment on WebSphere® Application Server, or on WebLogic. You do not need to deploy the Unica Interact design time environment. The design time environment is automatically deployed with the Campaign WAR or EAR file.

Reviewing and modifying the SQL upgrade script

If your Unica Interact runtime environment includes customizations to the runtime system tables that modified the default Data Definition Language (DDL) included with Unica Interact, you must modify the default SQL upgrade script for your database to match your customizations.

Common customizations include changes to support multiple audience levels or using views of tables. You can review the data dictionaries for the new versions of products to confirm that column sizes map correctly and that foreign key constraints from additional products do not conflict.

The aci_runtab_upgrd and the aci_usrtab_upgrd are the SQL upgrade scripts that most likely require revisions.

> ⚠️ **Important:** You must complete the changes before you run the Unica Interact upgrade tool.

Complete the following steps to review and modify the SQL upgrade script:

1. Locate the upgrade script for your database type. The scripts are installed in the `/ddl/Upgrades` or `/ddl/Upgrades/Unicode` directory under your Unica Interact installation after you run the Unica installer in upgrade mode.

2. Ensure that your database schema matches the Data Definition Language (DDL) included with Unica Interact. If your database schema does not match the DDL in the upgrade script, edit the script for your database type to match your environment.

   The following example shows the required modifications to the aci_runtab_upgrd SQL upgrade script to support the Household audience level:

   Your existing Unica Interact design time environment contains an additional audience level called Household. To support the Household audience level, your Unica Interact runtime environment database contains tables named HH_CHStaging and HH_RHStaging.

   Required changes to the upgrade script:

   a. Locate the code in the SQL upgrade script that updates the response history and treatment sizes for the Customer audience level and replicate it for your Household audience level. Change the table names in the SQL statements to the appropriate names for your Household audience level.

   b. You must also revise the SQL script to support the data type change for the SeqNum column in the UACI_RHStaging table. The value of the SeqNum is a sequential number across all response history staging tables. The next value that is used is tracked by the NextID column in the UACI_IdsByType table, where TypeID is 2. For example, you have three audience levels, customer, household, and account. In the customer response history staging table, the highest SeqNum is 50. In the household response history staging table, the

highest SeqNum is 75. In the account response history staging table, the highest SeqNum is 100. Therefore, you must alter the SQL to set the NextID for TypeID = 2 in the UACI_IdsByType to 101.

The following example SQL statements show the required additions to the aci_runtab_upgrd_sqlsvr.sql script for a SQL Server database that contains the Household audience level. The text that is added to support the Household audience level is in bold:

```
ALTER TABLE UACI_CHStaging ADD RTSelectionMethod int NULL
go


ALTER TABLE UACI_RHStaging ADD RTSelectionMethod int NULL
go


ALTER TABLE HH_CHStaging ADD RTSelectionMethod int NULL
go


ALTER TABLE HH_RHStaging ADD RTSelectionMethod int NULL
go


insert into UACI_IdsByType (TypeID, NextID) (select 2,
 IDENT_CURRENT('UACI_RHStaging') + IDENT_CURRENT('HH_RHStaging')
 + IDENT_INCR( 'UACI_RHStaging' ))
go


select * into UACI_RHStaging_COPY from UACI_RHStaging
go


select * into HH_RHStaging_COPY from HH_RHStaging
go


DROP TABLE UACI_RHStaging
go
```

```
CREATE TABLE UACI_RHStaging (

       SeqNum              bigint NOT NULL,

       TreatmentCode       varchar(512) NULL,

       CustomerID          bigint NULL,

       ResponseDate        datetime NULL,

       ResponseType        int NULL,

       ResponseTypeCode    varchar(64) NULL,

       Mark                bigint NOT NULL

                                       DEFAULT 0,

       UserDefinedFields   char(18) NULL,

  RTSelectionMethod    int NULL,

       CONSTRAINT iRHStaging_PK

              PRIMARY KEY (SeqNum ASC)

)

go


insert into UACI_RHStaging (SeqNum, TreatmentCode, CustomerID,

 ResponseDate,

 ResponseType, ResponseTypeCode, Mark, UserDefinedFields,

 RTSelectionMethod)

  (select SeqNum, TreatmentCode, CustomerID, ResponseDate,

 ResponseType,

  ResponseTypeCode, Mark, UserDefinedFields, RTSelectionMethod from

  UACI_RHStaging_COPY)

go


DROP TABLE UACI_RHStaging_COPY

go


DROP TABLE HH_RHStaging

go
```

```
CREATE TABLE HH_RHStaging (

        SeqNum                  bigint NOT NULL,

        TreatmentCode           varchar(512) NULL,

        HouseholdID              bigint NULL,

        ResponseDate            datetime NULL,

        ResponseType            int NULL,

        ResponseTypeCode        varchar(64) NULL,

        Mark                    bigint NOT NULL

                                        DEFAULT 0,

        UserDefinedFields    char(18) NULL,

   RTSelectionMethod     int NULL,

        CONSTRAINT iRHStaging_PK

               PRIMARY KEY (SeqNum ASC)

)

go


insert into HH_RHStaging (SeqNum, TreatmentCode, HouseHoldID,
 ResponseDate,
 ResponseType, ResponseTypeCode, Mark, UserDefinedFields,
 RTSelectionMethod)
  (select SeqNum, TreatmentCode, HouseHoldID, ResponseDate,
 ResponseType,
  ResponseTypeCode, Mark, UserDefinedFields, RTSelectionMethod from
  HH_RHStaging_COPY)
go


DROP TABLE HH_RHStaging_COPY
go
```

For DB2® and Oracle databases, the following statement is used for inserting values into the UACI_IdsByType table:

```
INSERT into UACI_IdsByType (TypeID, NextID)

 (select 2, COALESCE(max(a.seqnum)+1,1)

 + COALESCE(max(b.seqnum)+1,1)

 from UACI_RHSTAGING a, ACCT_UACI_RHSTAGING b );
```

If you have multiple audiences, you must add the following sections to the
aci_usrtab_upgrd SQL script for each audience level:

```
ALTER TABLE HH_ScoreOverride  ADD

        OverrideTypeID       int NULL,

        CellCode             varchar(64) NULL,

        Zone                 varchar(64) NULL

go


ALTER TABLE HH_ScoreOverride  ADD

        Predicate            varchar(4000) NULL,

        FinalScore           float NULL,

        EnableStateID        int NULL

go


CREATE INDEX iScoreOverride_IX1 ON HH_ScoreOverride
(

        HouseHoldID                       ASC
)
go
```

Setting environment variables

Set environment variables in the `setenv` file to upgrade the Unica Interact design time and
runtime environment.

Edit the `setenv` file to set the environment variables that are required by the Unica Interact
upgrade tools.

For the Unica Interact design time environment, the file is in the
Interact_Design_Environment_Install_Directory/interactDT/
tools/upgrade directory under the Unica Interact design time environment
installation. For the Unica Interact runtime environment, the file is in the
Interact_Runtime_Environment_Install_Directory/tools/upgrade directory
under the Unica Interact runtime environment installation.

For more information, read the comments in the setenv file.

The following table describes the environment variables that you must set for the Unica
Interact design time upgrade tools in the setenv file:

Table 8. Environment variables for the Unica Interact design time environment

**This two-columned table provides information about the names of the environment
variables in one column, and the description of the environment variables in the second
column.**

Variable	Description
JAVA_HOME	The root directory of the JDK used by your new Unica Campaign installation. For example, *<Installation_HOME>/ jre*
JDBCDRIVER_CP	The path to the directory that contains the JDBC driver. **JDBCDRIVER_CP** is the default path to the JDBC driver; you can override the path when you run the upgrade tool. Specify the same JDBC driver that was used while installing Unica Platform.
JDBCDRIVER_CLASS	The class for the JDBC driver. **JDBC-DRIVER_CLASS** is the default class to the JDBC driver; you can override the class when you run the upgrade tool.

Table 8. Environment variables for the Unica Interact design time environment

This two-columned table provides information about the names of the environment variables in one column, and the description of the environment variables in the second column.

(continued)

Variable	Description
JDBCDRIVER_URL	The URL for the JDBC driver. **JDBC-DRIVER_URL** is the default URL for the JDBC driver; you can override the URL when you run the upgrade tool.
ERROR_MSG_LEVEL	The desired logging level that has the following valid values, which are listed from most to least verbose: • DEBUG • INFO • ERROR • FATAL
LOG_TEMP_DIR	The directory where you want the migration tool to create the log files.
LOG_FILE_NAME	The name of the log file for the upgrade tool.

The following table describes the environment variables that you must set for the Unica Interact runtime upgrade tools in the `setenv` file:

Table 9. Environment variables for the Unica Interact runtime environment

This two-columned table provides information about the names of the environment variables in one column, and the description of the environment variables in the second column.

Variable	Description
JAVA_HOME	The root directory of the JDK used by your new Unica Interact installation.
JDBCDRIVER_CP	The path to the directory that contains the JDBC driver. **JDBC-DRIVER_CP** is the default path to the JDBC driver; you can override the path when you run the upgrade tool.
JDBCDRIVER_CLASS	The class for the JDBC driver. **JDBCDRIVER_CLASS** is the default class to the JDBC driver; you can override the class when you run the upgrade tool.
JDBCDRIVER_URL	The URL for the JDBC driver. **JDBCDRIVER_URL** is the default URL for the JDBC driver; you can override the URL when you run the upgrade tool.
ERROR_MSG_LEVEL	The desired logging level that has the following valid values, which are listed from most to least verbose: • DEBUG • INFO • ERROR • FATAL
LOG_TEMP_DIR	The directory where you want the migration tool to create the log files.
LOG_FILE_NAME	The name of the log file for the upgrade tool.

The environment variables for SSL upgrade are required for both the Unica Interact design time and runtime environments.

The following table describes the environment variables that you must set to support SSL upgrade for the design time and runtime environment:

Table 10. Environment variables to support SSL upgrade (runtime and design time environments)

This two-columned table provides information about the names of the environment variables in one column, and the description of the environment variables in the second column.

Variable	Description
IS_WEBLOGIC_SSL	Should the connection to the server of the target system be through SSL? The valid values are YES and NO. If the value is set to NO, you do not need to set the remaining SSL properties.
BEA_HOME_PATH	The path to the location where the WebLogic server of the target system is installed. You must point to the `license.bea` file in this path. If you install Unica Interact in a distributed environment where the WebLogic server of the target system is not available locally to the script, copy the `license.bea` file locally to some folder, and specify the path to that folder by using this environment variable.
SSL_TRUST_KEYSTORE_FILE_PATH	The path of the trust store that is used to configure SSL in the WebLogic server of the target system. The trusted certificates are saved at this location. The **SSL_TRUST_KEYSTORE_FILE_PATH** variable is used for SSL handshake.

Table 10. Environment variables to support SSL upgrade (runtime and design time environments)

This two-columned table provides information about the names of the environment variables in one column, and the description of the environment variables in the second column.

(continued)

Variable	Description
SSL_TRUST_KEYSTORE_PASS-WORD	The password of the trust store that is used to config-ure SSL in the WebLogic server of the target system. If there is no password, set it to `""` or nothing. The **SSL_-TRUST_KEYSTORE_PASSWORD** variable is used for SSL handshake.

Running the Unica Interact upgrade tools

Run the upgrade tool for the design time environment to update the Unica Interact tables in the Unica Campaign system tables. Run the upgrade tools for the runtime environment to update the Unica Interact run time, learning, contact history, response history, and user profile tables.

Running the upgrade tool for the design time environment

If you are upgrading from 12.1.0.3 to 12.1.0.4 or higher, you must specify the required values again in the setenv.sh/.bat file.

Before you run the upgrade tool, start the web application server on the target system.

The Unica Interact design time environment uses the Unica Campaign system tables as the database.

When running the upgrade tool for the design time environment, you can stop the upgrade at any prompt by typing `abort`.

The user who runs the upgrade tool must have access to the appropriate database client executable files (`sqlplus`, `db2`, or `osql`) for the Campaign system tables data source.

The latest version of the upgrade tool (aciUpgradeTool) is in the `/interactDT/tools/upgrade` directory under your Unica Interact design time environment installation. Enter the requested information at the prompts to upgrade your system tables for the new version of Unica Interact. When the tool completes successfully, your upgrade process is complete.

If you have multiple partitions, configure and run the upgrade tool once for each partition.

> **Note:** If audience specific Detail contact history tables (example:
> UA_DTLContactHist) are not mapped in Campaign configuration under
> the path `partitions|partition#|systemTableMapping` before
> upgrading environment for upgraded customers (if customers are upgrading
> to version 12.1.0.3 and onwards), then upgraded customers must add
> "ABTestBranchID" column in audience specific Detail contact history tables
> (example: UA_DTLContactHist) manually to view A/B Test Performance related
> informations. Upgraded customers can add this column by using the following
> query:
>
> - For Oracle DB:
>
>   ```
>   ALTER TABLE <audience specific Detail contact history table> ADD
>   ABTestBranchID NUMBER(19,0);
>   ```
> - For other DBs:
>
>   ```
>   ALTER TABLE <audience specific Detail contact history table> ADD
>   ABTestBranchID BIGINT;
>   ```
> - Example:
>
>   ```
>   ALTER TABLE UA_DTLContactHist ADD ABTestBranchID BIGINT;
>   ```

> **Note:** In case of a failure in migration of old strategies, you can use Strategy
> migration utility tool to migrate specific strategy to smart strategy after fixing the
> issue.

Running the upgrade tools for the runtime environment

If you are upgrading from 12.1.0.3 to 12.1.0.4 or higher, you must specify the required values again in the setenv.sh/.bat file.

Before you run the upgrade tools, start the web application server on the target system.

The Unica Interact runtime environment uses the Unica Interact system tables as the database.

When running the upgrade tools for the runtime environment, you can stop the upgrade at any prompt by typing `abort`.

The latest versions of the upgrade tools are in the `/tools/upgrade` directory under your Unica Interact runtime environment installation. Enter the requested information at the prompts to upgrade your tables for the new version of Unica Interact. When the tool completes successfully, your upgrade process is complete.

> ⚠️ **Important:** Run the SQL scripts/upgrade tools once for each server group.

Run the tools in the following order to upgrade the Unica Interact runtime environment:

1. Run aciUpgradeTool_runtab to update the systemTablesDataSource and the Unica Interact runtime configuration properties.
2. If you are using built-in learning, run aciUpgradeTool_lrntab to update the learningTablesDataSource.
3. If you are using cross-session response tracking, modify the `/tools/upgrade/conf/ACIUpgradeTaskList_crhtab.properties` file if necessary, and then run aciUpgradeTool_crhtab to update the contactAndResponseHistoryDataSource.

   You must modify the `ACIUpgradeTaskList_crhtab.properties` file if you are upgrading from Unica Interact version 8.x and if the Unica Interact runtime data source (as specified in the **contactAndResponseHistoryDataSource** configuration property under the **Interact | general** category) is not the same as the Unica Campaign system tables data source.

4. If you are using the scoreOverride or defaultOffers tables, run aciUpgradeTool_usrtab to update the prodUserDataSource.

> **Note:** If customers are having Platform-Campaign on one VM and Platform-Interact on second VM, then on Platform-Interact VM users must copy jdbc.properties from Platform-Campaign VM and replace by taking backup at the following locations `<Interact_Home>/tools/bin and <Install_Location>/install` and run `aciUpgradeTool_crhtab.sh`.
>
> After running aciUpgradeTool_crhtab.sh tool, the users must restore jdbc.properties file on Platform-Interact VM.

Redeploying the Unica Interact runtime server in the web application server

After you finish upgrading Unica Interact, redeploy the newly installed version of the Unica Interact runtime server in the WebSphere® Application Server, or on WebLogic. **NOTE:** After upgrade, it is observed that configuration Node treatmentStore is displayed under :Affinium| Interact|services|contactHist| Removal of this configuration can be done using the Platform configTool.sh/configTool.bat

Upgrade log

When you upgrade Unica Interact, the Unica Interact upgrade tools write processing details, warnings, and errors to the `aci_upgrade.log` file. Check the log file to verify that you have an error-free and clean upgrade.

By default, the name of the log file is `aci_upgrade.log` and the log file is in the `logs` directory, which is in the same directory as the Unica Interact upgrade tools. The location of the log file and level of verbosity are specified in the `setenv` file. You can modify the `setenv` file before you run the Unica Interact upgrade tools.

Upgrading partitions

For the design time environment, if you have multiple partitions, you must run the upgrade tool once for each partition. For the runtime environment, if you have multiple partitions, run the upgrade tool once on each runtime server.

Partitions must have the same names in the source and target versions of Unica Interact.

Creating and populating the Unica Interact system tables

If you have not created and populated the system tables during the installation process, use your database client to run the Unica Interact SQL scripts against the appropriate database or to create and populate the Unica Interact runtime environment, design time environment, learning, user profile, and contact and response tracking data sources.

Design time environment tables

Before you can enable the Unica Interact design time environment in Unica Campaign, you must add some tables to your Unica Campaign system table database.

The SQL scripts are in the *INTERACT_HOME*/interactDT/ddl directory under your Interact design time environment installation.

If your Unica Campaign system tables are configured for Unicode, use the appropriate script that is in the *INTERACT_HOME*/interactDT/ddl directory in your Unica Interact design time environment. There are no Unicode equivalent scripts for the aci_populate_systab scripts that are used to populate the design time environment tables.

Use the scripts in the following table to create the UnicaInteract design time environment tables:

Table 11. Scripts for creating design time environment tables

This two-columned table provides information about the data source type in one column, and the script name in the second column.

Data source type	Script name
DB2®	aci_systab_db2.sql The user table space and system temporary table space where the Unica Campaign system tables exist must each have a page size of 32K or greater.
Microsoft™ SQL Server	aci_systab_sqlsvr.sql
Oracle	aci_systab_ora.sql
MariaDB	aci_systab_mariadb.sql

Use the scripts in the following table to populate the Interact design time environment tables:

Table 12. Scripts for populating design time environment tables

This two-columned table provides information about the data source type in one column, and the script name in the second column.

Data source type	Script name
DB2®	aci_populate_systab_db2.sql
Microsoft™ SQL Server	aci_populate_systab_sqlsvr.sql
Oracle	aci_populate_systab_ora.sql
MariaDB	aci_populate_systab_mariadb.sql

Runtime environment tables

The SQL scripts are in the `<INTERACT_HOME>/ddl` directory under your Interact installation.

If your Interact runtime tables are configured for Unicode, use the appropriate script that is in the `<INTERACT_HOME>/ddl/Unicode` directory to create the runtime tables. There are no Unicode equivalent scripts for the aci_populate_runtab scripts that are used to populate the runtime tables.

You must run the SQL scripts once for each server group data source.

Use the scripts in the following table to create the Interact runtime tables:

Table 13. Scripts for creating runtime environment tables

This two-columned table provides information about the data source type in one column, and the script name in the second column.

Data source type	Script name
DB2®	aci_runtab_db2.sql The user table space and system temporary table space where the Interact runtime environment tables exist must each have a page size of 32K or greater.
Microsoft™ SQL Server	aci_runtab_sqlsvr.sql
Oracle	aci_runtab_ora.sql
MariaDB	aci_runtab_mariadb.sql

Use the scripts in the following table to populate the Interact runtime tables:

Table 14. Scripts for populating runtime environment tables

This two-columned table provides information about the data source type in one column, and the script name in the second column.

Data source type	Script name
DB2®	aci_populate_runtab_db2.sql You must use the following command when you run the script: db2 +c -td@ -vf aci_populate_runtab_db2.sql
Microsoft™ SQL Server	aci_populate_runtab_sqlsvr.sql
Oracle	aci_populate_runtab_ora.sql
MariaDB	aci_populate_runtab_mariadb.sql

Note: You should alter the size of the `UACI_EligStat.offerName` column from 64 to 130 (or 390 for Unicode tables) to preserve compatibility with Unica Campaign. Use the following sample SQL statements for this modification.

```
Non-Unicode

DB2: ALTER table UACI_EligStat ALTER COLUMN OfferName SET DATA TYPE
 varchar(130);
ORACLE: ALTER TABLE UACI_EligStat MODIFY OfferName varchar2(130);
SQLSVR: ALTER TABLE UACI_EligStat alter column OfferName
 varchar(130) not null;
MariaDB: ALTER TABLE UACI_EligStat modify column OfferName
 varchar(130) not null;
```

```
Unicode

DB2: ALTER table UACI_EligStat ALTER COLUMN OfferName SET DATA TYPE
 varchar(390);
ORACLE: ALTER TABLE UACI_EligStat MODIFY OfferName varchar2(390);
```

```
SQLSVR: ALTER TABLE UACI_EligStat alter column OfferName
 nvarchar(390) not null;
MariaDB: ALTER TABLE UACI_EligStat modify column OfferName
 nvarchar(390) not null;
```

Learning tables

You can use SQL scripts to create and populate tables for optional features such as learning, global offers, score override, and contact and response history tracking.

All the SQL scripts are in the *<Interact_HOME>*/ddl directory.

Note: The built-in learning module requires a separate data source from the Unica Interact runtime environment tables. For the built-in learning module, you must create a data source to hold all the learning data. The separate data source can communicate with all server groups, which means you can learn from your different touchpoints at the same time.

If your Interact runtime tables are configured for Unicode, use the appropriate script that is in the *<Interact_HOME>*/ddl/Unicode directory to create the learning tables.

Use the scripts in the following table to create the Interact learning tables:

Table 15. Scripts for creating learning tables

This two-columned table provides information about the data source type in one column, and the script name in the second column.

Data source type	Script name
DB2®	aci_lrntab_db2.sql
Microsoft™ SQL Server	aci_lrntab_sqlsvr.sql
Oracle	aci_lrntab_ora.sql

Table 15. Scripts for creating learning tables

This two-columned table provides information about the data source type in one column, and the script name in the second column.

(continued)

Data source type	Script name
MariaDB	aci_lrntab_mariadb.sql

Contact and response history tables

You must run SQL scripts against the contact history tables if you want to use cross-session response tracking or the advanced learning feature.

All the SQL scripts are in the Interact installation directory.

> **Note:** Using contact and response history features requires a separate data source from the Interact runtime environment tables. To use the contact and response history features, you must create a data source to reference contact and response data. The separate data source can communicate with all server groups.

If your contact history tables are configured for Unicode, use the appropriate script that is in the `Unicode` directory under the same location as the standard script to create the learning tables.

Use the scripts in the following table to create the Interact contact and response history tables:

Table 16. Scripts for creating contact history tables

This two-columned table provides information about the data source type in one column, and the script name in the second column.

Data source type	Script name
DB2®	• aci_crhtab_db2.sql in the `<Interact_HOME>/ddl/` directory. The script impacts the Interact runtime tables. • aci_lrnfeature_db2.sql in the `<Interact_HOME>/interact-DT/ddl/acifeatures/` directory. The script impacts the Campaign design time tables.
Microsoft™ SQL Server	• aci_crhtab_sqlsvr.sql in the `<Interact_HOME>/ddl/` directory. • aci_lrnfeature_sqlsvr.sql in the `<Interact_HOME>/interact-DT/ddl/` directory.
Oracle	• aci_crhtab_ora.sql in the `<Interact_HOME>/ddl/` directory. • aci_lrnfeature_ora.sql in the `<Interact_HOME>/interact-DT/ddl/` directory.
MariaDB	• aci_crhtab_mariadb.sql in the <Interact_Home>/ddl/ directory. • aci_lrnfeature_mariadb.sql in the <Interact_Home>/interactDT/ddl directory.

Deploying Unica Interact

You must deploy the Unica Interact runtime environment for every instance of the runtime server that you install. The Interact design time environment is deployed automatically with the Campaign EAR or WAR file.

You must know how to work with your web application server. Consult your web application server documentation for details.

HTTP proxy support in API and the OMO gateway

- When triggered messages are configured to use a HTTP connection, a HTTP proxy can be configured optionally with authentication between Interact and the endpoint.
- When the client library (interact_client.jar) is used to connect the client to Interact run time servers, an HTTP proxy can be configured optionally with authentication between the client application and Interact runtime.

Deploying the design time environment

After you install Unica Interact, the design time environment is deployed automatically when you deploy Unica Campaign. After you deploy the `Campaign.war` file, configuration procedures automatically enable the Unica Interact design time environment in Campaign. The `Campaign.war` file is in the Campaign installation directory.

Deploying the runtime environment

You must deploy the Unica Interact runtime environment by deploying the `InteractRT.war` file for every instance of the runtime server that you install or upgrade. For example, if six instances of a runtime server exist, you must install and deploy the Unica Interact runtime environment six times. You can deploy the runtime environment on the same server as the design time environment, or you can deploy the Unica Interact runtime environment on a separate server. The `InteractRT.war` is in the Unica Interact installation directory.

> 📝 **Note:** When you deploy the Unica Interact runtime environment, the context root must be set to `/interact`. Do not use any other value for the context root, or navigation to the runtime environment, and within Interact runtime links and pages, do not operate correctly.

> 📝 **Note:** If Unica Interact is upgraded it is required to set the INTERACT_HOME environment variable pointing to the Unica Interact installation directory to generate interact.log file. Release 11.1 onwards, the log4j is changed to 'log4j2'. You will need

> to copy any custom entries from the 'interact_log4j.peoperties' file of base setup to the 11.1 target setup in the 'interact_log4j2.xml'.

> **Note:** In case of deployment of Tomcat, include the following parameters in the `instance_name/conf/server.xml` file.
>
> `maxHttpHeaderSize="209715200"`
>
> `maxPostSize="-1"`

Deploying Unica Interact on WebSphere Application Server

You can deploy the product runtime environment on supported versions of WebSphere® Application Server (WAS) from a WAR file or EAR file. The design time environment is deployed automatically with the product EAR or WAR file.

- Ensure that multiple language encoding is enabled in WAS.
- When you run the Install New Application wizard, ensure that you set the JDK Source Level to 18.
- Ensure that you add javax.el-3.0.1-b11.jar in WAS server lib directory.

Deploying Unica Interact on WAS from an EAR file

You can deploy Unica Interact by using an EAR file if you included Interact in an EAR file when you ran the Unica installer.

- Confirm that your version of WebSphere® meets the requirements in the *Recommended Software Environments and Minimum System Requirements* document, including any necessary fix packs or upgrades.
- Confirm that you created the data sources and database provider in WebSphere®.

1. Go to the WebSphere® Integrated Solutions Console.
2. Complete the following steps, if your system tables are in DB2®:

a. Click the data source that you created. Go to the **Custom Properties** for the data source.

b. Select the **Custom properties** link.

c. Set the value for the **resultSetHoldability** property to 1.

   If you do not see the **resultSetHoldability** property, create the **resultSetHoldability** property and set its value to 1.

3. Go to **Applications > Application Types > WebSphere enterprise applications** and click **Install**.

4. In the **Preparing for the application installation** window, select the **Detailed - Show all options and parameters** check box and click **Next**.

5. Click **Continue** to see the **Install New Application** wizard.

6. Accept the default settings on the windows of the Install New Application wizard except the following windows:

   • In step 1 of the **Install New Application** wizard, select the **Precompile JavaServer Pages files** check box.
   • In step 3 of the installation wizard, set the **JDK Source Level** to 18.
   • In step 9 of the installation wizard, set the **Context Root** to /Interact.

7. In the left navigation panel of WebSphere® Integrated Solutions Console, navigate to **Applications > Application Types > WebSphere enterprise applications**.

8. In the Enterprise Applications window, select the EAR file that you want to deploy.

9. In the **Web Module Properties** section, click **Session Management** and select the following check boxes:

   • **Override session management**
   • **Enable Cookies**

10. Click **Enable Cookies**, and in the **Cookie name** field, enter a unique cookie name.

11. If you are using version 8 of WebSphere® Application Server, select **Servers > WebSphere application server > server 1 > Session management > Enable Cookies** and clear the check box for **Set session cookies to HTTPOnly to help prevent cross-site scripting attacks**.

12. In the **Detail Properties** section, select **Class loading and update detection**.

13. In the **Class loader order** section, select the **Classes loaded with parent class loader first** option.

14. In `campaign.ear`, open the **Manage Modules** section and select the **Classes loaded with local class loader first (parent last)** option.

15. For **WAR class loader policy**, select the **Class loader for each WAR file in application** option.

16. Go to **Application Servers > your server > Process definition > Java Virtual Machine**.

17. In the **Generic JVM arguments** section, enter the following JVM arguments:

```
-Dcom.ibm.websphere.webservices.DisableIBMJAXWSEngine=true

-Dibm.cl.verbose=PersistenceProvider

-Dibm.cl.verbose=PersistenceProviderImpl

-agentlib:getClasses -verbose:dynload

-Dcom.ibm.xml.xlxp.jaxb.opti.level=3
```

18. Start your deployment.

Deploying Unica Interact on WAS from a WAR file

You can deploy the Unica Interact application from a WAR file on WAS.

Complete the following tasks before you deploy the product:

- Confirm that your version of WebSphere® meets the requirements in the *Recommended Software Environments and Minimum System Requirements* document, including any necessary fix packs or upgrades.
- Confirm that you created the data sources and database provider in WebSphere®.

1. Go to the WebSphere® Integrated Solutions Console.

2. Complete the following steps if your system tables are in DB2®:

   a. Click the data source that you created. Go to the Custom Properties for the data source.

   b. Select the Custom properties link.

    c. Set the value for the **resultSetHoldability** property to 1.

       If you do not see the **resultSetHoldability** property, create the

       resultSetHoldability property and set its value to `1`.

3. Go to **Applications > Application Types > WebSphere enterprise applications** and
click **Install**.

4. In the **Preparing for the application installation** window, select the **Detailed - Show all
options and parameters** check box and click **Next**.

5. Click **Continue** to see the **Install New Application** wizard.

6. Accept the default settings on the windows of the **Install New Application** wizard
except the following windows:

    • In step 1 of the **Install New Application** wizard, select the **Precompile
JavaServer Pages files** check box.

    • In step 3 of the installation wizard, set the **JDK Source Level** to 18.

    • In step 9 of the installation wizard, set the **Context Root** to /Campaign.

    • In step 10 of the installation wizard, set the **Context Root** to /interact.

7. In the left navigation panel of WebSphere® Integrated Solutions Console, navigate to
Applications > Application Types > WebSphere enterprise applications.

8. In the Enterprise Applications window, click the `unica.war` file.

9. In the Enterprise Applications window, click the `Campaign.war` file.

10. In the Enterprise Applications window, click the `InteractRT.war` file.

11. In the Enterprise Applications window, click the `plan.war` file.

12. In the **Web Module Properties** section, click **Session Management** and select the
following check boxes:

    • **Override session management**

    • **Enable Cookies**

13. Click **Enable Cookies**, and in the **Cookie name** field, enter a unique cookie name.

14. If you are using version 8 of WebSphere® Application Server, select **Servers >
WebSphere application server > server 1 > Session management > Enable Cookies**
and clear the check box for **Set session cookies to HTTPOnly to help prevent cross-
site scripting attacks**.

15. In the **Applications > Enterprise Applications** section of the server, select the WAR file
that you deployed.

16. In the **Detail Properties** section, select **Class loading and update detection**.

17. In the **Class loader order** section, select the **Classes loaded with local class loader first (parent last)** option.

18. In the **Modules** section, select **Manage modules > interact**, and under **Class loader order** select the **Classes loaded with local class loader first (parent last)** option.

19. Enter the context root for the application as `/interact`.

20. In the **WAR class loader policy section**, select the **Single class loader for application** option.

21. To deploy Interact, select **interact > Manage Modules > interact.war > Class loader order**. For **Class loader order**, select **Classes loaded with local class loader first (parent last)** in **Manage Modules**.

22. For **WAR class loader policy**, select **Single class loader for application**.

23. Go to **Application Servers > your server > Process definition > Java Virtual Machine**.

24. In the **Generic JVM arguments** section, enter the following JVM arguments:

```
-Dcom.ibm.websphere.webservices.DisableIBMJAXWSEngine=true

-Dibm.cl.verbose=PersistenceProvider

-Dibm.cl.verbose=PersistenceProviderImpl

-agentlib:getClasses -verbose:dynload

-Dcom.ibm.xml.xlxp.jaxb.opti.level=3
```

25. Start your deployment.

Deploying Unica Interact on WebLogic

You can deploy Unica products on WebLogic.

Use the following guidelines when you deploy Unica Interact on WebLogic:

- HCL Unica products customize the JVM used by WebLogic. You might need to create a WebLogic instance that is dedicated to Unica products if you encounter JVM-related errors.

- Verify that the SDK selected for the WebLogic domain you are using is the Sun SDK by looking in the startup script (`startWebLogic.cmd`) for the JAVA_VENDOR variable. It should be set to: `JAVA_VENDOR=Sun`. If it is set to `JAVA_VENDOR=BEA`, JRockit has been

selected. JRockit is not supported. To change the selected SDK, refer to the WebLogic documentation.

- Deploy the Unica products as web application modules.

- On UNIX™ systems, you must start WebLogic from the console to allow correct rendering of graphical charts. The console is usually the machine on which the server is running. However, in some cases the web application server is set up differently.

  If a console is not accessible or does not exist, you can emulate a console using Exceed. You must configure Exceed so that your local Xserver process connects to the UNIX™ machine in root window or single window mode. If you start the web application server using Exceed, you must keep Exceed running in the background to allow the web application server to continue running. Contact Technical Support for detailed instructions if you encounter problems with chart rendering.

  Connecting to the UNIX™ machine via telnet or SSH always causes problems rendering charts.

- If you are configuring WebLogic to use the IIS plug-in, review the WebLogic documentation.

- Add the following parameters in the `JAVA_OPTIONS` section of `startWeblogic.cmd` or `startWeblogic.sh`:

  ```
  -Dinteract.home=Unica Interact installation directory
  -Dfile.encoding=UTF-8
  ```

- If you are deploying in a production environment, set the JVM memory heap size parameters to at least 1024 by adding the following line to the `setDomainEnv` script:

  Set `MEM_ARGS=-Xms1024m -Xmx1024m -XX:MaxPermSize=256m`

- Under certain circumstances, deploying older legacy interactive channels or interactive channels with large deployment histories can stress the system and require 2048mb or greater of Unica Campaign designtime and/or Interact runtime Java™ heap space.

  System administrators can adjust the amount of memory available to the deployment systems via the following JVM parameters:

-Xms####m -Xmx####m -XX:MaxPermSize=256m

Where the characters #### should be 2048 or higher (depending on their system
load.) Note that a 64-bit application server and JVM are usually necessary for values
greater than 2048.

These are the suggested minimum values. Analyze your sizing requirements to determine
correct values for your needs.

Deploying Unica Interact on Tomcat Application Server

You can deploy the Unica Interact WAR on the Tomcat Application Server (TAS).

Procedure to include during Tomcat Application Server configuration

When you configure Unica Interact on Tomcat Application Server you must perform the
following steps:

You must add Test, Production and Interact Runtime Data Source in Campaign.xml of the
Unica Campaign Tomcat instance. For example:

```
<Resource name="<testDataSource>"
 factory="com.unica.manager.tomcat.utils.TomcatDSFactory"
auth="Application" type="javax.sql.DataSource"
maxActive="30" maxIdle="10" maxWait="10000"
username="<db user for test schema>" password="<db password>"
 driverClassName=
"<db specific class name>" url="<db specific jdbc url>"/>


<Resource name="<prodDataSource>"
 factory="com.unica.manager.tomcat.utils.TomcatDSFactory"
 auth="Application" type="javax.sql.DataSource"
maxActive="30" maxIdle="10" maxWait="10000"
username="<db user for prod schema>" password="<db password>"
 driverClassName=
```

```
"<db specific class name>" url="<db specific jdbc url>"/>


<Resource name="<InteractRunTimeDataSource>"
 factory="com.unica.manager.tomcat.utils.TomcatDSFactory"
auth="Application" type="javax.sql.DataSource"
maxActive="30" maxIdle="10" maxWait="10000"
username="<db user for runtime schema>" password="<db password>"
driverClassName="<db specific class name>" url="<db specific jdbc url>"/>


<Resource name="<InteractLearningDS>"
 factory="com.unica.manager.tomcat.utils.TomcatDSFactory"
auth="Application" type="javax.sql.DataSource"
maxActive="30" maxIdle="10" maxWait="10000"
username="<db user for runtime schema>" password="<db password>"
driverClassName="<db specific class name>" url="<db specific jdbc url>"/>
```

```
Note  - {{You can encrypt DB password using  }}encryptTomcatDBPasswords
 utility located in <Platform_Home>/tools/bin
```

Note: Add prefix "`java:/comp/env/`" when defining datasource/JNDI.

For example: Runtime datasource must be given as "`java:/comp/env/InteractRTDS`" and similarly for other datasources: InteractLearningDS, prodUserDataSource, testRunDataSource, contactAndResponseHistoryDataSource.

Deploying Unica Interact on JBoss

You can deploy the Unica Interact application from a WAR file or an EAR file on the JBoss Application Server (JAS).

Consider the following points before you deploy Unica Interact on JBoss:

- Verify that your version of JBoss meets the requirements that are described in the *Recommended Software Environments and Minimum System Requirements* document, including any necessary fix packs or upgrades.
- You use the JBoss Integrated Solutions console to configure your JBoss Application Server. The steps that follow provide guidance for setting specific controls.

Note: Different versions of JBoss Application Server present user interface controls in varying sequences and with alternative labels.

Complete the following steps to set up the environment for deploying Unica Interact:

1. JVM settings can be added to the standalone.conf script in the `JBOSS_EAP_HOME/bin` folder.
2. If you are deploying in a production environment, set the JVM memory heap size parameters to at least 1024 by adding the following line to the `standalone.conf` script:`Set MEM_ARGS=-Xms1024m -Xmx1024m -XX:MaxPermSize=256m`
3. Under certain circumstances, deploying older legacy interactive channels or interactive channels with large deployment histories can stress the system and require 2048mb or greater of Campaign design-time and/or Interact runtime Java heap space.System administrators can adjust the amount of memory available to the deployment systems via the following JVM parameters:`-Xms####m -Xmx####m -XX:MaxPermSize=256m`where the characters #### should be 2048 or higher (depending on their system load.) **Note**: A 64-bit application server and JVM are usually necessary for values greater than 2048. These are the suggested minimum values. Analyze your sizing requirements to determine correct values for your needs.
4. Go to the JBoss Enterprise Application Platform Admin Console.
5. Deploy interact.ear /interactRT.war on JBOSS.

Configuring activityOrchastrator feature with JBOSS

To use activityOrchastrator feature please perform following steps.

1. Start JBOSS Standalone service with following command: standalone.bat –server-config=standalone-full.xml standalone.sh –server-config=standalone-full.xml
2. Open JBOSS CLI and connect to JBoss server and execute following commands:jms-queue add --queue-address=<queue-name> --entries=[java:/jms/<queue-name>] <queue-name>: Queue Name (For e.g. test)
3. Add application server user as well using following command from <JBOSS-HOME>/bin location by using: add-user.bat/sh -a -u <username> -p <password> -g guest For e.g. uname/pass : jboss/jboss123
4. Create JDBC connections as mentioned in JBOSS section of "Information for creating JDBC connections" in this guide.
5. Deploy <path-to-war>/interactRT.war
6. You needs to add two receiver parameters to configure activityOrchastrator - Username and Password .Note - [Username and Password is the credential of JBOSS application server]
7. Restart the server.

Verifying the Unica Interact installation

You must verify whether Unica Interact is correctly installed by confirming that you can access the interactive channels and the Unica Interact runtime URL.

1. To verify that the Unica Interact design time environment is installed, log in to the Unica console and confirm that you can access **Campaign > Interactive Channels**.
2. Complete the following steps to verify that the Unica Interact runtime environment is correctly installed:

   a. Access the Unica Interact runtime URL by using a supported web browser. The runtime URL is:

   ```
   http://host.domain.com:port/interact/jsp/admin.jsp
   ```

   where *host.domain.com* is the computer where UnicaInteract is installed and *port* is the port number on which the Interact application server is listening.

   b. Click **Interact Initialization Status**.

If the Unica Interact server is running correctly, UnicaInteract responds with the following message:

```
System initialized with no errors!
```

If the initialization fails, review the installation procedure and confirm that you followed all the instructions.

Additional configuration for enhanced security

The procedures in this section describe some additional configurations in the web application server. They are optional, but you might want to perform them for enhanced security.

Disabling the X-Powered-By flag

If your organization is concerned that the X-Powered-By flag in the header variable is a security risk, you can disable it using this procedure.

1. If you are using WebLogic, in the administration console, under ***domainName* > Configuration > Web Application**, set **X-Powered-By Header** to **X-Powered-By Header will not be sent**.
2. If you are using WebSphere®, perform the following steps.
    a. In the WebSphere® administration console, navigate to **Servers > Server Types > WebSphere application servers >** *server_name* **> Web Container Settings > Web container**.
    b. Under **Additional Properties** select **Custom Properties**.
    c. On the Custom Properties page, click **New**.
    d. On the Settings page, create a custom property named `com.ibm.ws.webcontainer.disablexPoweredBy` and set the value to `false`.
    e. Click **Apply** or **OK**.
    f. Click **Save** on the console task bar to save your configuration changes.
    g. Restart the server.

Configuring a restrictive cookie path

In your web application server, you can restrict cookie access to a specific application for enhanced security. If not restricted, cookies are valid across all the deployed applications.

1. If you are using WebLogic, perform the following steps.

    a. Extract the `weblogic.xml` file from the WAR or EAR package where you want to add the restrictive cookie path.

    b. Add the following to the `weblogic.xml` file, where *context-path* is the context path for the deployed application. For Unica applications the context path is typically /unica.

    ```
    <session-descriptor>

        <session-param>

          <param-name>CookiePath</param-name>

          <param-value>/context-path> </param-value>

        </session-param>

    </session-descriptor>
    ```

    c. Rebuild the WAR or EAR file.

2. If you are using WebSphere®, perform the following steps.

    a. In the WebSphere® administration console, navigate to the **Session Manager > Cookie** tab.

    b. Set the **Cookie Path** to the context path of your application.

    For Unica applications the context path is typically /unica.

Chapter 4. Uninstalling Unica Interact

Run the Unica Interact uninstallers to uninstall Unica Interact. When you run the Unica Interact 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.

In addition to the general instructions for uninstalling Unica products, follow the guidelines when you uninstall Unica Interact:

- If you have multiple Unica Interact runtime installations using the same Unica Platform installation, you must remove network connectivity for the Unica Interact runtime workstation before you run the uninstaller. Failure to do so uninstalls the configuration data for all the other Unica Interact runtime installations from Unica Platform.
- You can safely ignore any warnings about failure to unregister with Unica Platform.
- You can export a copy of your configuration as a precaution before uninstalling Unica Interact.
- If you uninstall the Unica Interact design time environment, after you run the uninstaller, you may need to manually unregister Unica Interact. Use the configtool utility to unregister `full_path_to_Interact_DT_installation_directory\interactDT\conf\interact_navigation.xml`.

> **Note:** On UNIX™, the same user account that installed Unica Interact must run the uninstaller.

1. If you have deployed the Unica Interact web application, undeploy the web application from WebSphere® or WebLogic.
2. Stop the Unica Interact listener.
3. Shut down WebSphere® or WebLogic.
4. Stop the processes that are related to Unica Interact.
5. 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.
6. Complete one of the following steps to uninstall Interact:
   - Click the Unica Interact uninstaller that exists in the `Uninstall_Product` directory. The uninstaller runs in the mode in which you installed Unica Interact.
   - In a command-line window, navigate to the directory where the uninstaller exists, and run the following command to uninstall Unica Interact 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 Interact:

     ./Uninstall_interact.

     > **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 Interact using the silent mode:

Uninstall_*Product* -i silent

When you uninstall Unica Interact 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 Interact, the Unica Interact uninstaller runs in the mode in which Unica Interact is installed.

Chapter 5. 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 17. Product names for configTool registration and unregistration

Product name	Name used in configTool
Unica Platform	`Manager`
Unica Campaign	`Campaign`
Unica Interact	`interact`
Unica Optimize	`Optimize`
Unica Plan	`Plan`

-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_12.0.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"
```