Reporting Guide for Cisco Unified Customer Voice Portal, Release 12.6(2)

First Published: 2023-04-28

Last Modified: 2023-04-28

Preface

Change History

Change	See	Date
Initial release of document for release 12.6(2)		April 2023

About this Guide

The *Reporting Guide for Cisco Unified Customer Voice Portal* provides the following information:

- Reporting service and reporting architecture.

- Cisco Unified Intelligence Center Reporting application and its various functions.

- Unified CVP templates for reporting.

- Database schema and database management.

- Guidelines for reporting.

Audience

This guide is intended for managers, Unified CVP system managers, Cisco Unified Intelligent Contact Management Enterprise (Unified ICME)/ Cisco Unified Intelligent Management Hosted (Unified ICMH) system managers, VoIP technical experts, and IVR application developers.

Related Documents

Unified CVP provides the following reporting related documentation:

- *Administration Guide for Cisco Unified Customer Voice Portal*

- *Solution Design Guide for Cisco Unified Contact Center Enterprise*

- *Feature Guide - Writing Scripts for Unified Customer Voice Portal*

- *Operations Guide for Cisco Unified Customer Voice Portal*

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.

- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.

- To submit a service request, visit Cisco Support.

- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.

- To obtain general networking, training, and certification titles, visit Cisco Press.

- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

Documentation Feedback

Provide your comments about this document to: mailto:contactcenterproducts_docfeedback@cisco.com.

CONTENTS

CHAPTER 3

CHAPTER 1

Reporting Server

Reporting Service

Reporting is an optional component for Cisco Unified Customer Voice Portal (CVP) installation.

Select **Reporting** during the installation process to install the Reporting Server, which is comprised of the reporting service and the reporting database.

The reporting service receives reporting data from the Interactive Voice Response (IVR) service, the Session Initiation Protocol (SIP) service (if used), and the VoiceXML (VXML) server and transforms and writes this data to the Informix reporting database to provide historical reporting in a call center environment.

Reporting data includes summary information about call activity, which assists call center managers in reviewing and managing daily operations. It can also include operational detail data for various IVR applications.

Reporting Architecture

The following diagram shows the Unified CVP architecture. For clarity, the diagram separates the reporting service and the database.

Note   The connection of the Operations Console to the call server through an OAMP Resource Manager (ORM) is simply indicative, because the ORM is invisible to the end user. An ORM is co-located with each managed Unified CVP component, and the Operations Console is connected to each component.

Figure 1: CVP Architecture
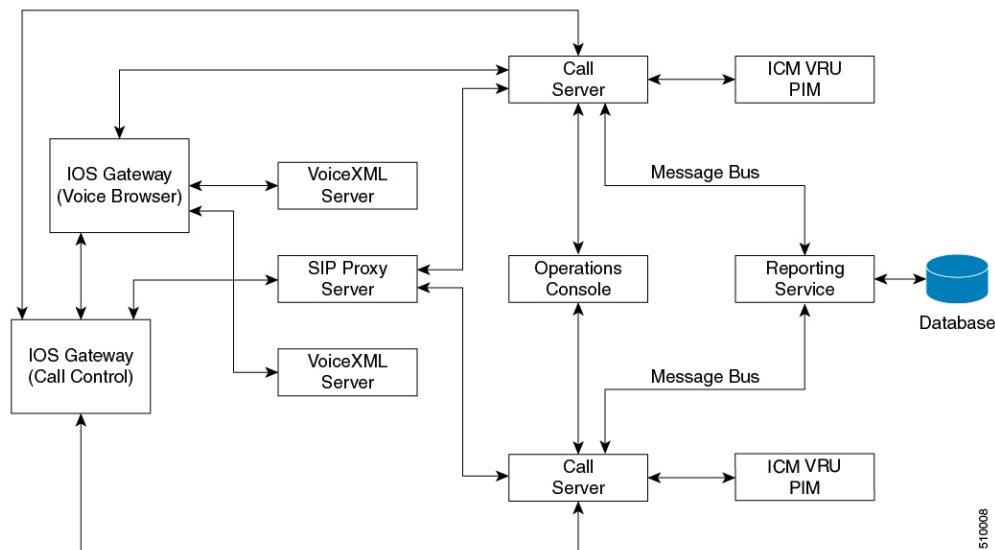
The call server uses a central messaging bus to allow each service to communicate.

The reporting service connects to the message bus through either an in-process plug-in or an out-of-process plug-in, depending on whether the reporting service resides in the same Java Virtual Machine (JVM) with the message bus system.

The service listens to all messages passing through the message bus and captures both call-state change messages sent from Call Server or reporting messages sent from the Reporting Server.

The reporting service then parses those messages, converts them into batches of applicable Structured Query language (SQL) statements, and runs them into an SQL database using the Java Database Connectivity (JDBC) Application provisioning interface (API).

The reporting service can also receive and process Unified CVP messages related to Unified CVP system administrative tasks, such as turning on or off debugging and querying statistics. As the Figure 1 shows, the reporting service can be shared by multiple Call Servers that belong to the same Unified CVP deployment.

Note A deployment needs only one reporting server. During temporary database outages, messages are buffered to file and are inserted into the database after the database comes back on line. The amount of time that messages can be buffered depends on the system throughput. See Reporting User.

If your environment uses more than one reporting server, be aware that:

- Each Call Server and each VXML Server can be associated with only one reporting server.

- Reports cannot span multiple Informix databases.

Although Unified CVP does not have a built-in reporting engine, its installation includes reporting templates designed for use with the Unified Intelligence Center (Unified IC) reporting application. You can import these templates into Unified IC and run them from the Unified IC interface.

Cisco Unified Customer Voice Portal Reporting Server Deployment Options and Sizing

You can find the Unified CVP reporting solution deployment options, together with related sizing requirements in the *Solution Design Guide for Cisco Unified Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-implementation-design-guides-list.html.

Cisco Unified Customer Voice Portal Reporting Server Installation and Upgrade

Explanations and procedures regarding the installation and upgrade of the Unified CVP reporting server are documented in the Installation and Upgrade Guide for Cisco Unified Customer Voice Portal.

Topics in the Installation and Upgrade guide include:

- Installing the reporting component
- Specifying the reporting password
- Excluding the reporting server from anti-virus software port blocking
- Upgrading the reporting server
- Adding reporting capability to the VXML Server
- Backing up and purging of the reporting database

Cisco Unified Customer Voice Portal Reporting Server Setup

You can find explanations and procedures regarding the configuration and maintenance of the Unified CVP Reporting server in the *Administration Guide for Cisco Unified Customer Voice Portal* and in the *Configuration Guide for Cisco Unified Customer Voice Portal*.

Topics in the operations console help and in the Configuration guide include:

- Reporting server statistics
- Adding a reporting server
- Editing a reporting server
- Deleting a reporting server
- Finding a reporting server
- Configuring a VoiceXML server for reporting (adding and editing)
- Applying inclusive and exclusive VoiceXML filters for reporting
- Transferring a file to multiple devices

Database Maintenance

Through the Operations Console, Unified CVP provides access to database maintenance and enables you to perform administrative tasks such as backups and purges.

See *Configuration Guide for Cisco Unified Customer Voice Portal* for details on database operations.

Cisco Unified Intelligence Center Reporting Application

Cisco Unified Intelligence Center Installation and Setup

Who can install and set up Unified Intelligence Center: any user in your organization.

Unified Intelligence Center is installed by DVD media on a Cisco Unified Voice Operating System (VOS) platform. VOS is an appliance or *closed box* model and does not support navigation into, or manipulation of, the file system.

Unified Intelligence Center can be installed as a standalone server or as a cluster of a maximum of eight server nodes. Unified Intelligence Center consists of one mandatory publisher node (called the *Controller*) and a maximum of seven subscriber nodes (called *Members*). The Controller node includes a Member; thus a deployment can consist of a Controller alone.

Figure 2: Unified IC Cluster
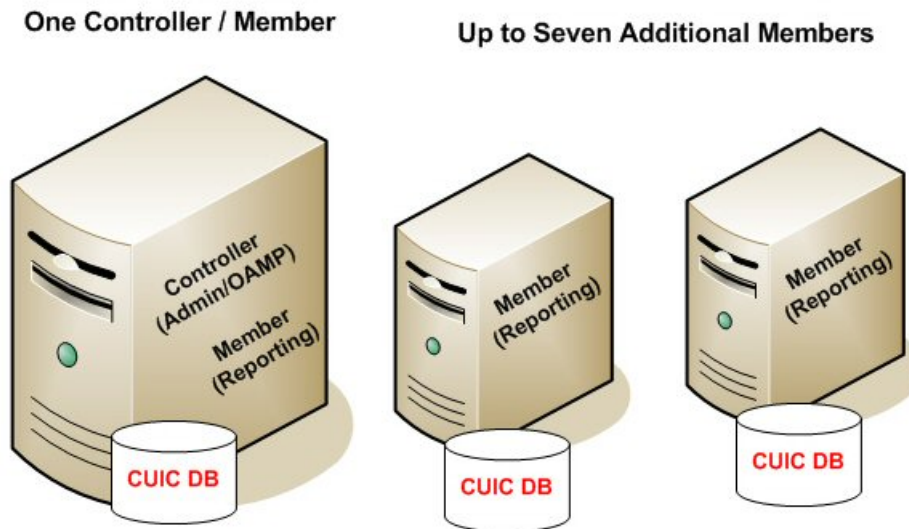
Unified Intelligence Center has two web interfaces:

- The **Administration** application

  The administration application is the **Operation Administration Maintenance and Provisioning** (OAMP) interface that provides application-level configuration for Unified Intelligence Center. This interface is offered on the Controller node only and is used by Unified Intelligence Center Super Users.

- The Unified Intelligence Center **Reporting** application

  The Unified Intelligence Center reporting application is the interface for reporting users who can have various user roles pertinent to reporting.

You can find the installation instructions in the *Installation and Upgrade Guide for Cisco Unified Intelligence Center* located at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-intelligence-center/products-installation-guides-list.html.

Access Administration Console

The System Application User who is defined during the installation is by default the initial Super User who can sign in to the Administration Application.

✎

Note   When you log in to the Admin Console site and you do not have any scheduled backup configured and enabled, Unified Intelligence Center returns the message "No active backup schedule is available. Set up a new schedule now." Unified Intelligence Center displays this message only for the administrator.

To access the Administration console:

Procedure

Step 1	Direct your browser to the URL *https://<HOST ADDRESS>/oamp* where *HOST ADDRESS* is the IP Address or Hostname of your server.
Step 2	Sign in using your Super User (system application user) ID and password. A successful sign-in launches the OAMP application.

Sign In to Cisco Unified Intelligence Center Reporting Interface

Who can sign in to the Unified Intelligence Center reporting interface:

  • Initially, the System Application User who is the default Superuser.

  • Eventually, any Unified CVP user who was created in the Administration Console as an IMS superuser or an LDAP user.

The URL for logging in to the Unified Intelligence Center reporting application is:

https://<HOST>:8444/cuicui/Main.jsp

Where HOST is the DNS name of a Unified Intelligence Center node.

> ✎
>
> **Note**   Unified Intelligence Center does not support HTTP.

Perform the following procedure to sign in to the Unified Intelligence Center reporting interface.

What to do next

If you have CVP Reporting as an on-box VM or an external server, refer to sections in the Configure Unified Intelligence Center section for information on creating the data source for Unified CVP and importing CVP report templates.

Create Reporting Users

Who can create a user:

  • Initially, the System Application User who is the default Superuser.

  • Eventually, any Superuser.

Unified CVP reporting users can sign in to Unified Intelligence Center only if they exist in the Administration console as Superusers or if Active Directory (AD) is configured in the Unified Intelligence Center Administration console for their domain:

  • Superusers who are added are considered to be IP Multimedia Subsystem (IMS) users.

  • Users who are authenticated through Active Directory are considered to be Lightweight Directory Access Protocol (LDAP) users.

Both IMS users and LDAP users can log in to Unified Intelligence Center reporting and are restricted to the limited Login User role until the Unified Intelligence Center reporting security administrator gives them additional roles and flags them as active users.

Create Superusers

Procedure

Step 1 Log in to the Cisco Unified Intelligence Center Administration Console (`https://<HOST ADDRESS>/oamp`).

Step 2 Navigate to **User Management**.

Step 3 Click **New** to add and configure a new user or click the ellipsis icon against an existing username to edit the configuration for that user.

This page has two tabs: **General** and **Credentials**. You can configure **Policy** information from the **User Management** page. For information about completing these tabs, see *Administration Console User Guide for Cisco Unified Intelligence Center* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-maintenance-guides-list.html or the Administration console online help.

Step 4 Click **Save**.

Create Reporting Users through OAMP

Procedure

Step 1 Log in to the Cisco Unified Intelligence Center Administration Console (`https://<HOST ADDRESS>/oamp`).

Step 2 Navigate to **Device Management > Unified CVP Reporting Server**.

Step 3 Select the Reporting Server for which a reporting user is to be created.

Step 4 Navigate to **Database Administration > Manage Reporting Users**.

Step 5 Click **Add User** to add and configure a new user or click an existing username to edit the configuration for that user.

Step 6 Click **Add** to add a new user.

All reporting users are created with Read Only privilege. Elevated privileges can be added with SQL DBA commands.

Configure Active Directory Settings

Fields on the Active Directory tab configure the Active Directory server to authenticate reporting users as they log in to the Unified Intelligence Center Web application.

Configure Active Directory for the Unified ICM/CC supervisors so that they can sign in as Unified Intelligence Center Reporting users.

Note Cisco Unified Intelligence Center uses LDAP V2 which does not support all Unicode characters that are used in the first name or surname of LDAP users.

Active Directory is not used to authenticate Administration Super Users. These Super Users can only be authenticated through the local database. The first Super User is added during installation. All other Super Users are added through the **User Management** interface, and their credentials are encrypted into the local database.

To navigate to this page, choose **Cluster Configuration** > **Active Directory Settings**.

Table 1: Fields on This Tab

Field	Description
Host Address and Port for Primary Active Directory Server	Provide the hostname or IP address and the port of the Primary Active Directory server.
	The port defaults to 389.
Host Name and Port for Secondary Active Directory Server	Provide the hostname or IP address and the port of the Secondary Active Directory server.
	The port defaults to 389.
Use SSL	Check these boxes if you want the connection from the Unified device to the Active Directory connection to be encrypted with SSL while doing authentication.
Manager Distinguished Name	Enter the Manager Distinguished Name used to sign in to the Active Directory server.
	For example, on a default installation of Microsoft AD: `CN=Administrator, CN=users, DC=MYSERVER, DC=COM`. Replace *MYSERVER* and *COM* with your respective hostname.
	Note If users other than the LDAP administrator are configured as Manager Distinguished Name in the OAMP LDAP configurations, they should have the following rights: 1. User search permissions on the domain. 2. Read access to the user objects and their attributes. 3. Read access to the base DN 4. Permission to bind to LDAP.
Manager Password	Enter the Active Directory manager password.

Field	Description
User Search Base	Specify the user search base. For example, on a default installation of Microsoft AD, `CN=users, DC=MYSERVER, DC=COM`, replace *MYSERVER* and *COM* with your respective hostname. **Note** This example assumes that you placed the users in the USERS subtree of AD. If you created a new organizational unit within your subtree, then the syntax would be: `OU=MYUSERS, DC=MYSERVER, DC=COM`, instead of "CN=MYUSERS".
Attribute for User ID	Whenever a user signs in, Unified Intelligence Center searches for that user in the LDAP (Lightweight Directory Access Protocol) using the sign-in attribute specified in the LDAP configuration. After the user is found, the full DNS of the user is extracted and used for authenticating the user. The sign-in attribute specified in the LDAP configuration is the property against which LDAP search is issued to find the matching username. If you do not know which attribute to use, use *sAMAccountName*, which is the default Microsoft username attribute. Different organizations settle on different LDAP attributes to identify the username across the organization, depending on the tools used to administer LDAP within their organizations. This attribute allows you to customize the sign-in depending on the attribute used. Even a custom attribute can be specified using this dialog. *sAMAccountName* indicates the user attribute to search the user for is the *userPrincipalName*. *sAMAccountName* contains just the short username. For example, jDoe for the user John Doe. *userPrincipalName* indicates the user attribute to search the user for is the *userPrincipalName*. This attribute contains the username in the email format, `user@compay.com`. This entire string becomes the username and not just user. Therefore when this attribute is selected, the user has to type the full email format in as the username in the sign-in box. **Custom User Attribute** allows you to specify the attribute used for searching the user in LDAP. **Note** Custom User attributes are not validated and are used as is. Ensure that the correct case and attribute name are used. Contact your Active Directory Administrator for the correct attribute to use.

Field	Description
User Name Identifiers	Users are stored in Unified Intelligence Center in the format <UserName Identifier>\<username>
	The username Identifiers are used to identify the different kinds of users within Unified Intelligence Center. For example, local, LDAP, user-synced user, users from different LDAP domains, nETBIOSName, and so on.
	Before you can use it, the username identifier has to be declared for use on this page. When LDAP is configured, at least one identifier must be configured and set as default to enable the system to identify LDAP users.
	UserSychronization brings in users in format <syncdomain>\username and collections have users in the same format. Therefore, these users must sign in to Unified Intelligence Center using the <syncdomain>\user syntax. To enable, add <syncdomain> or @<syncdomain> (if you are using *userPrincipalName*) to the list of valid identifiers.
	The maximum allowed length of a username identifier is 128 characters.
	Example:
	When nETBIOSName and userPrincipalName are same or different:
	For sAMAccountName:
	Configure in Username Identifiers: <UserNameIdentifier>
	Login in cuic : UserNameIdentifier\user
	For userPrincipalName:
	Configure in Username Identifiers: @<UserNameIdentifier>
	Login in cuic : userPrincipalName
	This list box is pre-populated with the username Identifiers based on the list of usernames stored in the Unified Intelligence Center database. The most frequently occurring identifier in the list of username is auto-selected as the default.
	Note You cannot save LDAP configuration unless you choose a default Identifier from the **User Name Identifiers** list box and clicking the **Set Default** button.
Default User Name Identifier	Default identifiers allow users to sign-in without typing the full domain identifier (<domain>\user) or the *userPrincipalName* suffixes to usernames (user <@company.com>) on the sign-in page.
	It can be set by choosing one of the Identifiers from the list box and by clicking the **Set Default** button.
	Users who use any other identifier can still sign-in by typing their full identifier in the sign-in box. For example, domain2\user or netbiosname\user, provided those identifiers are already configured.

Field	Description
Test Connection	Click to test the connection to the primary and secondary LDAP servers and display the connection status.

- **Save** saves the configuration information you entered for the Active Directory. Clicking **Save** *does not validate the configuration*.

Cisco Unified Customer Voice Protocol Reporting User Role Additions

Who can add Unified CVP user roles:

- Initially, the system application user who has full permissions in Unified Intelligence Center reporting

- Eventually, any security administrator.

- Initially, the system application user (see *System Application User*) who has full permissions in Unified Intelligence Center reporting

- Eventually, any security administrator (see *User Roles*).

Once Unified CVP users log in to Unified Intelligence Center, they are added to the Unified Intelligence Center database and appear on the users list.

New users are initially defined as Login Users: the lowest level user role of Unified Intelligence Center. A Unified Intelligence Center Security Admin user must access the **Users** page to check if **User is Active** and to grant additional user roles to the user.

Create Data Source

You can create or edit a data source only if you are assigned with a System Configuration Administrator role.

To create a data source, perform the following steps:

Procedure

Step 1   In the left navigation pane, choose **Configure** > **Data Sources**.

Step 2   In the **Data Sources** window, click **New**.

Step 3   In the **Create Data Source** dialog box, enter the datasource **Name**, **Description**, and select the **Data Source Type**.

Step 4   Click **Next**.

Step 5   In the data source details page, enter the following (Primary Node tab):

Field	Description
Host Settings	

Field	Description
Datasource Host	The hostname or IP address of the target data source.
Port	The port number that allows Unified Intelligence Center to communicate with the database. Typically, the port is 1526. **Note** The port number is a mandatory field only for the Informix database.
Database Name	Enter the name of the database. The database name can be *cvp_data* or *callback*.
Instance	Enter the instance of the database. By default, this is *cvp*. **Note** The name of the database instance is a required field only for Informix databases.
Time zone	Select the time zone that the database is located in. Set CVP datasource timezone configuration to UTC on CUIC.
Authentication Settings	
Database User ID	The user ID required to access the database. (The cvp_dbuser account is created automatically during Unified CVP Reporting server installation.)
Password	The password for the user ID required to access the database.
Charset	The character set that is used by the database. Choose UTF-8.
Max Pool Size	The maximum pool size. **Note** Value ranges from 5 to 200. The default Max Pool Size value is 100 and is common for both the primary and secondary data source tabs.

Step 6   Click **Test Connection** to ensure that the database is accessible and the credentials provided are correct.

Step 7   Click the **Secondary Node** tab to configure a failover for the data source.

Step 8   Check the **Enable Failover** check box to configure a failover for the data source.

Step 9   Enter the required details for the failover data source. (Refer step 5)

Step 10   Click **Save**.

Obtain Cisco Unified Customer Voice Portal Report Templates

Who can obtain import Unified CVP report templates: any user in your organization.

To import Unified CVP report templates complete the following:

Procedure

Step 1	On the Unified CVP Reporting Server, click **Start**.
Step 2	In the search box, type **%CVP_HOME%\CVP_Reporting_Templates** and press **Enter**.
Step 3	Compress the reports into a zip folder and copy it to the system from which you will run Unified Intelligence Center Administration.
Step 4	Choose the files and copy them to the client computer from which you will launch the Unified IC Reporting web application.

Import Reports

You can import the Unified Intelligence Center report, which is in either .xml or .zip file format.

To import reports, perform the following steps:

Procedure

Step 1	In the left navigation pane, choose **Reports**.
Step 2	In the **Reports** listing page, click **Import**.
Step 3	Click **Browse** to select the file (.xml or .zip format) to be imported.
	Note Maximum file size for .zip file format is 60 MB and for .xml file format is 3 MB.
Step 4	Select the required file and click **Open**.
Step 5	Select the file location from the **Save to Folder** list to save the file.
Step 6	Click **Upload**. Once the file is successfully uploaded, the table gets populated with the corresponding report template, current available version, and incoming version of the files being imported.
Step 7	Select a Data Source for the Report Definition only if the Report Definition for the report being imported is not defined in Unified Intelligence Center.
Step 8	Select a Data Source for the Value List that is defined in the Report Definition.
	Note Selection of a Data Source for the Value List is mandatory: • If the Value List does not use the same Data Source as the Report Definition. • For Real Time Streaming Report Definitions.

Step 9   Select the files to import or overwrite.

• Overwrite—If the report being imported exists in the Unified Intelligence Center.

• Import—If the report being imported is the new set of report files.

Step 10   Click **Import**.

Note   • If the system does not have report definitions, new report definitions are created in **Report Definitions** > **Imported Report Definitions** folder.

• Importing a report to a different version of Unified Intelligence Center is not supported. However, when you upgrade Unified Intelligence Center, report templates continue to work in the upgraded version.

• Importing manually edited XMLs is not supported.

• Templates will not be imported if there is no connection to the correct database in CVP Reporting Server.

Imported Cisco Unified Customer Voice Protocol Report Templates

Unified Intelligence Center has stock report templates and custom report templates.

Stock report templates are the UCCE templates that are installed with Unified Intelligence Center. These templates are populated from the Unified ICM/CC database. They are protected from modification and must be cloned using "Save As".

Custom report templates are templates that are imported. Unified CVP templates, therefore, fall in the custom template category.

Unified Intelligence Center does not protect custom templates from modification.

Retain the imported Unified CVP templates and create Save As versions of the Unified CVP reports and Unified CVP report definitions.

Export Reports and Folders

You can export any custom or stock report or report folders from the Unified Intelligence Center. Reports and report folders are exported in a ZIP file format.

When you export a folder, the reports in the folder are grouped as ZIP files. The grouping is based on the data source that is used by the report definition and the value lists.

Note

- For customized reports, you must update the version numbers of the value list and report definition before you export the report. Else, the export will not overwrite the existing reports.

- While exporting folders, ensure that all the Value Lists in the report definitions or folders point to the same data source respectively.

Caution Do not modify the exported report (XML file) for customization purposes. However, if necessary, you can modify only the EntityVersion of the Report, Report Definition, and ValueList.

When you export a report, the following data that is associated with the report are exported:

- Report
- Report Definition
- Value Lists
- Views
- Thresholds
- Drilldowns
- Template Help (if not bundled, an empty folder is created in the zip file)

Note Report Filters and Collections are not exported along with the report.

To export a report or a folder, perform the following steps:

Procedure

Step 1 From the left navigation panel, click **Reports**.

Step 2 Navigate to the report or the report folder that you want to export and click the ellipsis icon beside the report and click **Export**.

Note If necessary, you can rename the report or the report folder. Do not change the file extension (Reports or report folder: zip).

Step 3 Click **OK**.
The exported file is downloaded into your specified local folder.

Basic Cisco Unified Intelligence Center Reporting Concepts

The following table provides information about basic Unified Intelligence Center reporting concepts.

Table 2: Basic Unified Intelligence Center Reporting Concepts

Term	Explanation
Dashboards	Dashboard is an interface that allows you to add reports, web pages (URLs), web widgets, and notes in a consolidated view. All actions on the Dashboards interface are based on your role and on the user permissions for Dashboards and for Folders.
Data Sources	A data source is a connection to a database from which reports are populated. Each data source has a configuration page with the IP address, username, password, and database type for a database used by Unified Intelligence Center.
Reports	Reports contain data sets that are extracted by database queries and that can be displayed in various views--as grids, as charts, and as gauges.
	All reports are based on Report Definitions.
Report Definitions	A Report Definition defines the interface for a report. Each Report Definition contains the dataset that is obtained for a report including the SQL query, the fields, the filters, the formulas, the refresh rate, and the key criteria field for the report.
	Unified Intelligence Center separates Reports from Report Definitions.
Report Templates	Report Templates are well-formed XML files based on Report Definitions.
Report Views	A report view is a layout presentation for the data that is retrieved for the report. Unified Intelligence Center supports two types of views:
	• Grid Views
	• Chart Views
	All reports have a grid view. The Unified CVP Call Traffic reports also have a chart view.
	You can create many views for a report, can define the default view for a report, and can change a report view once the report is generated.
	You cannot delete all views. Every report must have at least one view.

Term	Explanation
Value Lists	Value lists contain all reportable items of the same type, for example, all agents or all skill groups.
Collections	Collections are subsets of value lists that can be used to control the amount of data that users can select to populate a report.
Thresholds	You can set a threshold for a field in a report grid to configure that field to display in a distinctive format.
User Groups	User Groups are constructs that allow security administrators to partition Unified Intelligence Center functionality. Creating User Groups expedites the provisioning process when multiple users need the same access to dashboards and reports, or when users require distinct permissions and features based on regional or organizational requirements.
User Permissions	Users have permissions associated with the groups in which they are members, and each member of a group has specific permissions in that group. There are two levels of permissions: VIEW and EDIT.
User Roles	User Roles confer the actions and capabilities that a user has in Unified Intelligence Center. There are seven User Roles, and each user can have multiple roles.

Generate and Manage Reports

Overview

Reports show data returned by Report Definitions (database query). The database queries extract this data and can be displayed in various Report Views—as grids and charts.

Users with the Report Designer user role can click the **Reports** from the navigation page to open the Reports page.

Note All actions on the Reports interface are based on user role and on the user's object permissions for reports and for folders.

For more information about creating or editing a report, see *Create Reports*.

Stock Reports

Cisco provides stock report templates to use with Unified Intelligence Center. You can download Stock reports from Cisco.com. Stock report templates display data that has been saved in the Unified CCE database.

After installing Unified Intelligence Center, you can import stock templates using the **Import** feature and customize the stock reports to suit your business requirements. You can clone the imported stock templates using the **Save As** option and customize.

Stock reports have one default grid view. Few stock reports also have a chart view.

Note   Localization of stock report templates is not supported.

The available stock report templates are:

- **Audit Trail**—view the sequence of audit records of the transactions related to create, update, modify, and delete that are performed on the entities of a Unified Intelligence Center server.

  By default, only System Administrators can access and view this report. System Administrator can give permissions to other Unified Intelligence Center users to use this report.

- **Group Access Detail**—view access rights of groups that use Unified Intelligence Center resources such as Dashboards, Reports, and other resources.

- **Resource Ownership and Access**—view access rights of users and about ownership status of users who use Unified Intelligence Center resources such as Dashboards, Reports, and other entities.

- **User's Audit Log**—view the audit log data of users such as, Logged In User Detail (name and role), Event Detail, Updated Column, User Detail, and other modified details.

For more information, click the "?" icon (Template Help) from the report in the run mode.

Report Actions

The following table lists various actions that you can perform from the Reports.

Note   You can open a maximum of ten tabs at a time.

Table 3: Report Actions

Action	Description
Report-level actions	
New	
Report	Creates a new report in the selected folder.
	Note You cannot create reports inside the Stock folder. You can only import reports into the Stock folder. To edit or customize reports, clone the report and edit the cloned version.

Action	Description
Folder	Creates a new Folder. Use this feature to categorize reports.
	Note When you move or save the folders to a different location, the drop-down lists all the folders including the disabled folders. You can navigate to subfolders with **Edit** permissions.
	Note You cannot create folders inside the Stock folder.
Toolbar actions	
Refresh	Refreshes the Reports page.
	Applies to all folder levels (root, sub folder, and report).
Favorites	To easily access your reports, you can tag Reports as your Favorites.
	Click the star icon beside the Report name to add to Favorites.
Search	Searches for a particular the Report.
Import	Imports a report.
	To import a report, you need the Report Designer, Report Definition Designer, and Value List Collection Designer roles and the **Edit** permission on the target folder where you want to import these reports.
	For more information, see *Import Reports*.
	Note Applies to all folder levels (root, subcategory, and report).
Ellipsis(…) actions	
Edit	Edits the Report details. In the edit mode, you can add, modify, and delete report details, views and thresholds, and filters.
	After editing the Report, click **Finish**.
	Note You cannot edit a Stock report.

Action	Description

Save As	Saves a copy of the report with a different name.
	
	Note • By default, the reporting users do not have permission to create a subfolder in the Reports root folder. To get permissions, contact your administrator.
	
	  • You cannot perform the **Save As** action to move contents (reports or folders) into the Stock folder and its subfolders.
	
	Note The report description does not support the following special characters:
	  • Parentheses (( ))
	  • Angle brackets (<,>)
	  • Forward slash (/)
	  • Question mark (?)
	  • Quotes (")
	  • Any scripts; JavaScript

Clone Report Definition	If you want to create a copy of the Report Definition that is associated with the report being saved: 1. Click the **Clone Report Definition** check box. 2. Enter the new **Report Definition Name** and select the Report Definition Location. 3. Click **Save**. The new report gets associated to the cloned Report Definition.
Rename	Renames a folder or a report. **Note** You cannot rename a Stock folder or a Stock report. **Note** Applies to the root-level folder.
Move	Moves Report or Folder from one folder to another. **Note** • You can move a Report or a Folder only if you have **Edit** permission on the parent folder of the Report or Folder being moved. • You cannot move custom folders or reports from within the Stock folder (and its subfolders) to other locations and the other way.

Action	Description
Set Default Filters	Creates report filters. For more information, see *Report Filters*. **Note** — You can also set the default filter by checking the **Set as Default** check box in the **Choose Filter** dialog box during the report run mode.
Add Help	Hosts the help page for Report Templates. For more information, see *Add Template Help*.
Delete	Deletes a report or a folder. **Note** • You can delete a Report or a Folder only if you have **Edit** permission on the parent folder of the Report or Folder being deleted. • You cannot delete a Stock folder or a Stock report.
Permissions	Assigns appropriate permissions to access and manage the Report. **Groups**—Grants **View** and **Edit** permissions for the Report. • Security Administrators can grant these permissions to various groups. • Entity owners can grant these permissions to groups that they are directly associated with. **Users**—Grants **View** and **Edit** permissions for the Report to various users. Applicable only to Security Administrators. **Note** • Higher permissions (View and Edit) from either an individual user or the user group takes precedence. • Only the first 200 records (alphabetical order) are displayed in the **Members** or **Groups** panel. To view more records, see **Configure** > **Groups**. • When you modify a permission and want to switch between **Groups** and **Users** tabs, you will be prompted to either save or discard the changes.
Permalinks	Displays the Report permalink. **Note** — You can access permanent hyperlink only from a web browser.
Export	Export any custom report or report folders. Reports and report folders are exported in a ZIP file format. To export a report or a report folder, you need the REPORT DESIGNER role. For more information, see *Export Reports and Folders*.

Add Template Help

You can configure individual help files to each Cisco Unified Intelligence Center report. You can either host the help page separately and point the report to it or create and upload the help page along with the report.

This help content is specific to the report and can contain explanation on:

- How to use the report
- Field description
- Details of the relationship between the fields
- How to interpret the report data or
- Any other report related information

You can upload only files in ZIP formats. ZIP files can contain multiple HTML files. The HTML page contents support rich text including images.

Note   Help files does not support videos and other interactive content.

To configure the help page for a report, perform the following steps:

Procedure

Step 1	From the left navigation pane, click **Reports**.
Step 2	Click the Ellipsis icon (…) next to the report row for which you want to create the help page and click **Add Help**.
Step 3	In the **Add Help** dialog box,

- If you want to set an external help page as the report help, select the **URL** option and enter the external URL location.

- If you want to upload the help file, select the **Choose file** option and click **Browse** to upload a ZIP file (with HTML files).

Step 4	After uploading the file, click **Save**.

Note   When you run the report, click the "?" icon (Template Help) on the Reports toolbar to view the configured help file.

Report Filters

Filter Types

Report filters in Unified Intelligence Center are used to present selective data. You can define the filter to filter the data that you want to display in the report. There are two ways to view the Filter page.

• Before the report is generated: You can set and refine the default filter values using the **Actions** > **Set Default Filter** option.

• After the report is generated: You can refine the filter values using the filter icon.

> **Note** You cannot view filters if the Report Designer has selected the **Skip filter during the report processing** check box during the report filter selection.

Cisco Unified Intelligence Center supports the following types of report filters:

• Date & Time

• Key Criteria

• Field Filters

• Parameters

Filter parameters are displayed based on the selected query type in the Report Definition for that report.

Table 4: Filter tabs

Report Definition Query Type	Applicable Filter Tabs
Database Query	Date & Time, Key Criteria, Field Filters
Live Data or Real Time Streaming	Key Criteria, Field Filters
Anonymous Block	Parameters
Stored Procedure	Parameters

Date & Time

> **Note** • Cisco Unified Intelligence Center uses the browser locale to display the Date & Time format in the filter page. If Cisco Unified Intelligence Center does not support the browser locale language, then the locale selected in the Cisco Unified Intelligence Center application is used.

To configure Date and Time filters for a report, perform the following steps:

Procedure

Step 1 After creating the report, click **Set Default Filter** from the ellipsis **Actions**.

Step 2 In the **Date & Time** filter wizard, select the **Date Range** and **Time Range** options.

• The options available in the **Date Range** and **Time Range** filter are predefined.

• Selection of the **Custom** option allows you to customize the **Date Range** and **Time Range** details.

You can select the days of the week (Days > Custom) only if the time interval spans more than a day.

For reports that are based on the query type Anonymous Block, you cannot select days of the week. For more information, see *Cisco Unified Intelligence Center Report Customization Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-intelligence-center/products-user-guide-list.html.

Step 3 Check the **Skip filter during the report execution** check box if you want to skip filter selection during the running of report.

> **Note** In the **Choose Filter** dialog box (report run mode), you can check the **Set as Default** check box to set the report filter as the default.

Step 4 Click **Next**.

Key Criteria

Use the **Key Criteria** tab in the filter to select value lists or collections. A collection is a pre-configured group of values.

> **Note** Key Criteria fields are predefined filters and are displayed in the filter screen if they are defined in the Report Definition.

To configure Key Criteria filters, perform the following steps:

Procedure

Step 1 After creating the report, click **Set Default Filter** from the ellipsis **Actions**.

Step 2 In the **Date & Time** filter wizard, select the **Date Range** and **Time Range** options and click **Next**.

Step 3 In the **Key Criteria** filter wizard, select the collections or values from the **Available** selection box.

> **Note** Do not set multiple filter parameters; set any one filter parameter.

Step 4 Use the arrows to move the selected collections or values to the **Selected** selection box.

Step 5 You can also select multiple collections or values.

Step 6 Click **Next**.

Field Filters

Use the **Field Filters** tab to filter any field in the report. Based on the selected field type (date, numeric/decimal, boolean, or string), different operators are available. For example, you can filter calls in queue for greater than two minutes or on all agents in the hold state to filter out the less important information.

Note You can configure Field Filters in reports:

- Only for SQL Query based reports.

- To filter a text, date, boolean, or a decimal field.

To configure Field Filters, perform the following steps:

Procedure

Step 1 After creating the report, click **Set Default Filter** from the ellipsis **Actions**.

Step 2 In the **Date & Time** filter wizard, select the **Date Range** and **Time Range** options and click **Next**.

Step 3 In the **Key Criteria** filter wizard, select the collections or values from the **Available** selection box and click **Next**.

Step 4 In the **Field Filters** wizard, select the filter according to the following criteria.

Filter criteria/operators depend on the selected field type (date, numeric/decimal, boolean, or string).

- For **Date**, the options available in Date Range filter are predefined. Using the Date Range drop-down list, select from Custom, Today, Yesterday, This Week, Last Week, This Month, Last Month, Year to Date, or Last Year.

  Only Custom will allow the user to customize the Calendar, Time Range and Days certain days of the week.

- For **Decimal**, select an Operator from Equal To, Not Equal To, Less Than, Less Than or Equal To, or Greater Than and then enter a value; for example, Operator = Greater Than and Value = 16.5.

- For **String**, select an Operator from Equal To, Not Equal To, or Matches and then enter a value for the string; for example, Operator = Matches and Value = Team Green.

    - If you select Matches as the Operator, you must specify an SQL pattern to match the string field. The system appends the wild card character % automatically to the beginning and end of the string. You can also use any SQL wild card pattern in between the string.

    - If the filter field is associated with a **Value List**, then specify any value or move one, all, or some items in the list to the Selected column to filter.

  Note   Use only the Matches operator to filter the report data records with leading or trailing spaces. Do not use the Equal To or Not Equal To operators in such scenarios as Unified Intelligence Center trims the filter criteria before running the SQL query and hence does not fetch the matching results.

- For **Boolean**, select True or False from the Operator list.

Step 5 Using the **Operator** drop-down list, select the criteria.

  Note   If you select **Matches** operator, you can use any Microsoft SQL wildcard pattern to filter the data. The wildcard character **%** is added to the beginning and end of every string that is used to filter the data.

| Step 6 | In the **Value** field, enter a value against which the data in the field will be filtered. |
| Step 7 | Click **Run**. |

Report Types

There are three types of reports based on the query types:

- Historical report

- Real-time report

- Live Data report

Report Components

These reports consist of the following components:

- **Data Source**

  The Data Source defines the sources that contain the data for the report. Unified Intelligence Center supports three types of data sources: Microsoft SQL Server, IBM Informix, and Streaming. The data source should be pre-configured for you. If it is not, contact your administrator to configure the appropriate data source.

- **Report Definitions**

  Each report has a Report Definition, which represents how data is retrieved from the data source for that report template. In addition to specifying how data is retrieved (by a simple MS SQL query, stored procedure query, real time streaming or an anonymous block query), a report definition contains the dataset that is obtained. This includes the fields, filters, formulas, refresh rate, and key criteria field for the report.

- **Reports**

  Reports show data returned by Report Definitions. This data is extracted by database queries.

- **Time Zone Conversions**

  The time zone conversion happens if there is a difference between the user and the data source time zones.

  Daylight savings time consideration for Database Query Reports: The Daylight savings time offset at the start of the date range is considered for the time zone conversion when the report is filtered. If the daylight savings change occurs somewhere in between the filter date and time ranges, the time zone offsets will not be computed properly if the user and the data source are in different time zones. In this case, you will have to split the time filter such that separate report invocations are run before and after the daylight savings time change.

> **Note** The daylight savings time offset depends on the latest system time zone library.

Note Daylight savings start date or end date may (when the data is recorded) create an additional row (in the report output) due to the time zone value change in the SQL database. This applies to any template report, as well as for any query type (Anonymous block, Database Query, Stored Procedure, Real Time Streaming) that these template reports would use. For confirmation that SQL had time zone value change, use the Report Options icon from the report summary page and run the SQL command in AW node to validate the change in the time zone.

Example 1:

```
User time zone: Australia/Sydney

Data source time zone: America/New_York

Daylight savings time changes: +1 hour for Sydney at 2013-10-06 02:00
a.m. DST +1 hour is already on for New York

Filter selected by the user: 2013-10-06 to 2013-10-06, 12:00 a.m. to
11:59 p.m.

Query formed in data source time zone: 2013-10-05 10:00:00 a.m. to
2013-10-06 09:59:59 a.m.

Report displayed in user time zone: 2013-10-06 12:00 a.m. to 2013-10-07
12:59 a.m.
```

In this case, the report will display an extra hour of data as Sydney Daylight savings time is off at the start of the date range and on at the end of the date range.

Example 2:

```
User time zone: Australia/Sydney

Data source time zone: America/New_York

Daylight savings time changes: +1 hour on for Sydney at 2013-10-06
02:00 a.m. DST +1 hour is already on for New York

Filters selected : 2013-10-06 to 2013-10-10, 03:00 a.m. to 11:59 p.m.

Query formed in data source timezone: 2013-10-05 12:00:00 p.m to
2013-10-06 08:59:59 a.m.

Report displayed in user time zone: 2013-10-06 03:00 a.m. to 2013-10-06
11:59 p.m.
```

In this case, the conversion happens as expected as there is no Daylight savings time change in between the date ranges.

Example 3:

```
User time zone: America/New_York

Data source time zone: IST

Daylight savings time changes: +1 hour DST at 2013-03-03 02:00 a.m.
and off at 2013-11-03 02:00 a.m. for New York. No Daylight savings
time changes for IST.
```

```
Filter selected by the user : 2013-11-03 to 2013-11-03, 01:30 a.m. to
05:30 p.m.

Query formed in data source time zone: 2013-11-03 11:00:00 a.m. to
2013-11-03 02:59:59 a.m.

Report displayed in user time zone: 2013-11-03 01:30 a.m. to 2013-11-03
04:30 p.m.
```

In this case, the report will display one hour less of data as New York Daylight savings time is on at the start of the date range and off at the end of the date range.

- **Report Views**

  A report can be presented in multiple formats (grid and charts). Each view can have its own set of fields. A single report can have multiple views.

- **Report Help**

  You can attach a help page specifically for your report. For more information, see *Add Template Help*.

Historical and Real Time Reports

Historical report—Retrieves data from the historical data source. Reports are populated with interval data that has a default refresh rate of 15 minutes. Historical reports have an upper limit of 8000 rows.

Real-time report—Retrieves data from the real-time data source. Reports are populated with interval data that has a default refresh rate of 15 seconds. Real-time reports have an upper limit of 3000 rows.

While running Cisco Unified Intelligence Center Historical and Real-Time Reports you can:

- Filter data in a report

- Change the view of a report from a grid to a gauge or a chart

Note You can select one of the predefined grid, gauge, or chart views.

For more information on creating/editing views, see Report Views.

- Modify reports

Live Data Reports

Live Data report—Receives data from streaming data source. Live Data reports supports only grid view.

Note Live Data reports do not automatically respond to changes in the system time. If the server or client time is changed or adjusted, the report must be refreshed to accurately display the duration field values. For example, during a daylight saving time (DST) change, active live data reports do not display correct values in the duration field. Live data reports must be refreshed to update.

Manage Reports

Create Reports

All actions on the Reports interface are based on user role and on the user's object permissions for reports and for categories.

Note By default, reporting users do not have permission to create a subfolder in **Reports**. An Administrator can create a subfolder and grant access.

To create a new report, perform the following steps:

Procedure

Step 1 In the left navigation pane, choose **Reports**.

Step 2 Navigate to the folder where you want to create the report.

Step 3 From the **Reports** toolbar, click **New** > **Report**.

To edit an existing report, navigate to the report, click the ellipsis icon beside the report and click **Edit**.

Step 4 In the **Create New Report** window, enter the Report **Name** and **Description**.

Note The report name must be unique to Cisco Unified Intelligence Center.

Step 5 Click **Next**.

Step 6 In the **Basic Details** tab, enter or select the report details.

Note Use the arrows to select the appropriate Report Definition.

Step 7 Click **Next**.

Step 8 In the **Manage Views** tab, create the report views and click **Next**.

For more information, see Report Views.

Step 9 In the **Thresholds** tab, create report thresholds and click **Finish**.

For more information, see Report Thresholds.

The newly created report is listed in the **Reports** page.

Report Views

There are two types of report views:

- Grid View

- Chart View

Note Do not delete the Report Definition fields that are currently associated with any of the manually created report views. If deleted, you must reapply the Data Fields for all the manually created report views to save the report.

For default grid views, the deleted field is removed automatically from the Data Fields list retaining any other fields in the Selected Field list. Hence, no additional action is required to save the report.

Create a Grid View

Grids are tabular presentations of the data in rows and columns. By default, all Cisco stock reports have a grid view. For custom reports, a default grid is created from the SQL query in the Report Definition.

Note Grouping and font size is not supported in Live Data reports.

You can create a Grid View while creating or editing a report.

To create a Grid View, perform the following steps:

Procedure

Step 1 Create or edit a report.

Step 2 Enter the report details in the **Basic Details** screen and click **Next**.
The **Manage Views** screen appears with a default grid view.

Note You can access Report permalinks only after completing the report creation. Report permalinks allow you to share your report with other users and view reports of other users.

Step 3 You can edit the default view (**Actions** column > **Edit View**) or click **Create New** > **Grid view** to create a new grid view.

The **Edit Grid View** or the **New Grid View** screen appears depending on your selection for edit or create.

Step 4 Enter the **Name** and **Description** in the respective fields.

Note Maximum length allowed for the grid view **Name**: 50 characters.

Step 5 From the **Font** selection box, you can select the font size from the list to display the grid data.

Step 6 Use the arrow buttons to select fields from the **Available** value list box to move to the **Selected** field list.

Step 7 You can use the following features to improve grid view display:

- **Header**—Use this feature to add (+) or delete (-) a header for the selected fields. This helps in categorizing the field set.

  Note You cannot save the view with empty headers.

  Post upgrade to Cisco Unified Intelligence Center 12.0 or later, any empty headers that exist in the report views in prior releases are not migrated.

- **Edit** icon—In the Selected value list box, click the **Edit** icon (hover on the field value) if you want to edit the **Display Name** and **Column Width** for the selected field and click **Done**.

> **Note** For Header fields, you can only edit the Display Name.

- **Sort Grid by Field**—Select the **Sort Grid by Field** check box to sort the selected report columns in either **Ascending** or **Descending** order. Selecting this check box enables the drop-down list to be populated with the values from the **Selected** value list box. You can select only one value for sorting.

Step 8 Click **Save**.
The **Report Views** screen appears.

Step 9 Click **Finish**.

Create a Chart View

Cisco Unified Intelligence Center supports the following chart types:

- **Bar**—Bar charts display discontinuous events and show the differences between events rather than trends. Bar charts are oriented vertically and can be stacked horizontally or clustered one below the other.

- **Pie/Donut**—Pie charts display quantities as proportions of a whole. The circle (pie) represents 100% of the data, with each quantity represented as a wedge of the appropriate size. Pie charts take decimal or numeric fields only. A pie chart cannot have more than 50 wedges. An error occurs if your data set and chart editor selections generate a pie chart with more than 50 wedges.

  A doughnut chart is another display representation of a pie chart.

- **Column**—Column charts display discontinuous events and show the differences between events rather than trends. Column charts are oriented horizontally and can be stacked vertically or clustered side by side.

- **Dial Gauge/Numeric**—A gauge chart displays the dial representation of the report results as per the defined threshold.

  The Numeric chart displays the report results in a number format highlighted as per the defined threshold.

- **Line charts**—Line charts display continuous quantities over time against a common scale. Use the Line charts to show trends.

> **Note**
> - Live Data reports do not support chart view.
>
> - In the vertically oriented charts, for Cyrillic characters, the data labels in the Horizontal Axis field may be hidden or garbled. This is a known limitation. Hence, for Cyrillic characters, use the horizontally oriented charts.

To create a Chart View, perform the following steps:

Procedure

Step 1 Create or edit a report.

Step 2 Enter the report details in the **Basic Details** screen and click **Next**.
The **Manage Views** screen appears with a default grid view.

Note	You can access Report permalinks only after completing the report creation. Report permalinks allow you to share your report with other users and view reports of other users. For more information, see *Permalink for a Report*.

Step 3   Click **Create New** > **Chart view**.

Step 4   In the **Create New Chart View** screen, click the required chart type. For more information, see *Chart Types*.

Step 5   Enter the Chart Information; **Name**, **Description** and click **Next**.

Note	Maximum length allowed for the chart view **Name**: 50 characters.

For Cartesian type charts (Bar, Column, and Line), select the **Group Data** check box to group data:

- **By a field**—Select this option to create a chart view where the vertical axis shows fields with footer formula configured for line or column chart and horizontal axis with footer formula for bar chart.

- **By label field**—Select this option to create a chart view where the vertical axis shows fields of decimal data type for Line or Column chart. In Bar chart, the horizontal axis shows fields of decimal type.

Note	For Pie charts, you can only **Group Data by Label Field**.

Step 6   In the **Add Data Fields** screen, select the **Label Field** from the drop-down list and **Data Fields** from the list box and click **Next**.

Step 7   In the **Preview and Format** screen, enter or select appropriate information based on the selected chart type.

For more information, see *Chart Types*.

Note	For the following Data Fields, the Column Type (Stacked and Grouped) feature is unavailable.

- Date and Time

- Boolean

Step 8   Click **Save**.

Chart Types

Chart Type	Chart Information	Add Data Fields	Preview and Format
Bar	Yes	Yes	Yes
Column	Yes	Yes	Yes
Line	Yes	Yes	Yes

Chart Type	Chart Information	Add Data Fields		Preview and Format
Gauge/Numeric	No	Yes		Yes

		Note	To configure a Gauge chart, Report Definition must have at least one decimal field with footer configured.	You can select **Dial Gauge** or **Numeric** view for this report.
				
				To set the chart view for Dial Gauge/Numeric, perform the following steps:
				
				1. Enter the Range (min and max). Default: 0-100
				
				2. Define the zones.
				
				   When the chart value is within any of the defined thresholds,
				
				     • The gauge pointer points to the corresponding color set in the threshold.
				
				     • The Numeric text is displayed in the corresponding color set in the threshold.
				
				3. Click **Save**.

Pie	Yes	Yes		Yes
		Note	To configure a Pie chart, Report Definition must have at least one decimal field configured.	You can select **Pie** or **Donut** as the display type for this report.

Report Thresholds

You can set a threshold indicator for a field to display if the field value meets the threshold condition. There are nine colors instead of the color palette for the threshold color selection in this release. Threshold indicators can be set only for view type **Grid** and **Chart** > **Gauge**.

For setting field threshold indicators for a **Chart** > **Gauge** view, see Create a Chart View.

For setting field threshold indicators for a **Grid** view, perform the following steps:

Procedure

Step 1	From the **Manage Views** screen, after adding the report views, click **Next**. The **Thresholds** screen appears.
Step 2	Select a view to which you want to set the threshold and select the field name from the **Create new threshold** list. The screen refreshes with a new panel for the selected field name.
Step 3	Select a field operator and set a condition from the Operator list.

Operator	Description
Matches	The **Matches** operator accepts Regular Expressions. Note that the Regular Expressions does not support: • Flags (i, g, m, n, y), OR/AND any combinations of these flags. • Leading and trailing forward slash (/). Example: • Valid Pattern → \w+\s • Invalid Pattern → /\w+\s/g (As it contains leading and trailing forward slash (/) and a "g" flag.)
String fields; Always, Equal, Not Equal Decimal fields; Always, Equal, Not Equal, Greater Than, Less Than, Greater Than Equal To, Less Than or Equal To, Between	In Report Definition, if the %format is defined for any field, then while setting the thresholds for that field, ensure to enter the decimal format of the percentage to render the condition in the report. For example: In Report Definition, if %format is defined for the field "SL" (Service Level) and you want to apply thresholds to this field to indicate "Red" if SL is less than 60%, set the following: **a.** Define the threshold for the SL field. **b.** Set the **Operator** to Less Than. **c.** Enter the percentage value as `0.60`. **d.** Select "Red" in the **No Fill** drop-down. **e.** Click **Done**.

Step 4	Choose the options from **No Fill** and edit the threshold fields.

Note	You can set conditions on the same or different fields:

- condition on same field: threshold and condition on the same field.

- condition on different field: threshold for a field, based on the condition on the different field.

- multi conditions on same field: apply threshold for a field based on the condition on different fields.

Caution	When you upgrade to Unified Intelligence Center version 11.6 or later, all the threshold colors are retained for reports that are created in the earlier versions. But, when you modify the threshold, all the old threshold color selection are lost within the report. Hence, you must reconfigure the threshold color selection for that report.

For existing reports, perform the above mentioned steps to add more thresholds.

Note	Threshold configuration supports upto 30 thresholds for a field.

To edit an existing threshold from a report run already, click **Report options** and select **Manage Thresholds.**

Step 5 Format the text in the field to appear when it matches the threshold condition. Use the following options:

- **Text Bold**—Select this check box to highlight the report field in bold.

- **Text/Background Color**—Select a color from the drop-down for the text/background color in the field.

- **Text Substitute**—Enter a new string if you want the text in the field to be replaced with it when it matches the threshold condition.

  - Syntax to add an html hyperlink as text substitute: *cisco*

  - Syntax to add an empty space as text substitute: * *

- **Image Location**—Enter the URL path of the image if you want the text to be replaced with an image.

Note	Supports only image URLs reachable from Unified Intelligence Center server. Maximum size limit allowed for the image is 5MB.

Step 6 Click **Done**.

Step 7 Click **Finish**.

Run Report

Report Actions - Run Mode

The following table lists all the menu items and actions you can perform when you run a report.

Table 5: Report Actions

Action	Description
Report options	
Edit View	Displays the **Edit View** dialog box. You can modify the current report view and click **Done** to instantly view the modified view.
Save View As	Clones the existing report view. In the **Save View** dialog box, enter the Name and Description for the cloned view and click **Save**. The report page refreshes with the cloned view.
Create Chart View	For the report that is run, you can directly create a chart view if you have **Edit** permissions. After you create the chart view, the report page refreshes with the newly created chart view and gets listed in the view list. **Note** This feature is disabled for Live Data reports. For more information, see *Create a Chart View*.
Group By	Add/remove/update grouping configurations for the current view (columns). Cisco Unified Intelligence Center grid reports support up to three levels of grouping. If you are grouping the column with Date or Date time data type, you can group records on a Daily/Weekly/Monthly basis. For more information, see *Group By*.
Manage Thresholds	Sets a threshold indicator for a field to display if the field value meets the threshold condition. Threshold indicators can be set only for views of type Grid and Gauge. For more information, see *Report Thresholds*.
SQL	Displays the SQL code used to run this report.
Export	Exports the already run grid report data into your local disk in a .xlsx format. **Note** • When you export a report to an Excel file format, to read the exported report, the client system's locale must match with the browser's locale (where you had exported the report). • When reports are exported to excel in Report viewer, the custom formatting of DECIMAL data type is not applied.
Report menu	
Run or Pause the report	Click to run or pause the report respectively. **Note** Report running times out after three minutes. Rerun the report by modifying the filter, and if the problem persists, contact your administrator.
Print Report	Prints the report using your default printer. **Note** Reports in chart view supports only landscape mode in A3 size paper for printing.

Action	Description
Manage Filters	Displays **Choose Filter** dialog box to modify filter criteria for this report. For more information, see *Report Filters*.
Refresh	Refreshes the Report page.
View Filter Information	Displays the filter information of the report.
Online Help	Displays the configured template help. You can configure template help for the report from the **Reports** page > **Add Help**. For more information, see *Add Template Help*.
Only Thresholds	Enable this toggle button to view only rows with matching threshold values in the report. By default, this check box is unchecked for every report. **Note** This button is disabled for the grouped view.

Group By

For a report that is run, use the **Group By** option to add/remove/update grouping configurations for the current view. Cisco Unified Intelligence Center grid reports support up to three levels of grouping.

If you are grouping the column with Date or Date Time data type, you can group records on a Daily/Weekly/Monthly basis.

Note Live data reports do not support grouping.

To group the report data, perform the following steps:

Procedure

Step 1 From the report that is run, click the **Report options** icon and select the **Group By** option.

Step 2 In the **Group By** dialog box, specify the **Number of Levels** you want to group the report.
Depending on the number of levels selected, the Level, Grouped By, Sub Group, and Show Expanded columns are activated.

Cisco Unified Intelligence Center grid reports support up to three levels of grouping.

Step 3 To group the report data by values in a particular column, select the required column name from the **Grouped By** list.

If you select a date or date and time value from the list, you can select any one of the following from the **Sub Group** column:

- None—The report data is grouped by the absolute date or date time values.

- Daily—The report data is grouped by day.

> • Weekly—The report data is grouped by week.
>
> • Monthly—The report data is grouped by month.

By default, the **Show Expanded** option is selected and you can uncheck the option if necessary. The **Show Expanded** column allows you to view the reports that are run with the group expanded.

Enable the **Show Summary Only** toggle button to display only the summary row in the report.

For example, if you group by Agent Team and enable the **Show Summary Only** toggle button, only the summary data row for each team is displayed.

> **Note**   If any of the fields have a footer formula defined in the report definition, then a group level summary is also displayed for such fields using that formula.

Step 4   Click **Save**.

> **Note**   • For the grouped view, the **Only Thresholds** check box is disabled.
>
>   • Thresholds are not displayed in a grouped field and on summary rows.
>
>   • You cannot perform a drill-down from a report with grouped fields.

Cisco Unified Intelligence Center Reporting User Roles

Your User Role allows you to access the application functionality that corresponds to that role.

With appropriate User Role, you can:

> • Assign a user to one or more of the six User Roles.
>
> • Assign multiple User Roles to individual users depending on the size, staff, geographical distribution, and security practices of your call center.
>
> • Distribute each user role to many users.

> **Note**   Role changes to a user who is currently signed in, must sign out and sign in again for the changes to take effect.
>
> When you modify the user roles, the changes are logged in the **CCBU-cuic.<timestamp>.startup.log** file with `ROLES_AUDIT_LOG` as prefix to the changes.

> **Caution**   Do not modify the user information of a user who is currently signed in, as the user will be automatically signed out.

Note The Login User role, earlier identified for all the users who could sign in to Unified Intelligence Center is now integrated within the system. To activate or inactivate a user, you can use the toggle button in the **Edit User** > **User Information** tab.

An active login user can:

- Sign in to Unified Intelligence Center.

- Access **Configure** > **Users** and edit their information. For example, to change their Alias or Last Name.

User Role	Supported Functions
System Configuration Administrator	Manages Data Sources and Scheduler.
Security Administrator	The initial, default Security Administrator is the user who is defined as the System Application User during the installation.
	Manages Users, User Groups, and User Permissions.
	Also, System Administrators can:
	• Assign User Roles—User Roles are associated with people. User Roles are assigned to users to control access to various functions and what objects you can create.
	• Assign users to User Groups.
	• Assign Permissions—Permissions are associated with objects (Dashboards, Reports, Report Definitions, Data Sources, Value Lists, and Collections).
	• Use the **Run As** feature to verify other users' permissions.
Dashboard Designer	Manages Dashboards.
Value List Collection Designer	Manages Value Lists and Collections.
Report Designer	Manages Reports. Can access Scheduler to work on reports with appropriate permissions.
Report Definition Designer	Manages Report Definitions.

Note For users who have been synched into Unified Intelligence Center from Unified CCE or Packaged CCE, you cannot edit the Report Designer and the Dashboard Designer roles.

Users in the Administration Console

There are three user accounts that have access to the Administration console:

- Super Users
- System Application User
- System Administration User

Super Users

This user role is defined in the Administration console. It is the only user role for Administration.

The initial and default Super User is the *System Application User* who is configured during installation.

The initial Super User (the System Application User) can sign in to the Unified Intelligence Center Reporting console and has *all* User Roles and *full permissions* for all drawers in Unified Intelligence Center Reporting. Those credentials cannot be removed from the initial Super User.

Additional Super Users who are added in the Administration Console can also sign in to Unified Intelligence Center Reporting and are considered to be IMS users. They have limited Login User role only, until the Unified Intelligence Center Reporting security administrator gives them additional roles and flags them as Active users.

Local users can log in using their IMS username and password. After logging in for the first time, the users are listed on the User List Page. The username is not case sensitive, but the password is case sensitive.

System Application User

This user role is defined during installation. Although it is possible to define unique application user names and passwords during the installation of each node, you must use the same credentials for all installations.

The application user defined during the installation of the *Controller* is the only System Application User recognized by Unified Intelligence Center.

This user has full rights to all functions in the Administration and Unified Intelligence Center Reporting applications, as described below:

- Can log in to the Administration application and becomes the initial Super User for Administration.
- Can create additional Super Users in the Administration application.
- Can log in to Unified Intelligence Center and has full rights to all functions in Unified Intelligence Center.
- Is the initial Security Administrator user in the Unified Intelligence Center Reporting application.
- Can create additional Security Administrator users in the Unified Intelligence Center Reporting application.
- Cannot have any role taken away from them.
- Cannot take any role away from himself.
- This user can log in to the Reporting application and is the initial System user.

System Administration User

The System Administrator account User ID and password are configured at installation for each node. You must enter the same user name and password for all nodes.

The System Administrator for the *Controller* can access:

- The Cisco Systems tools on the Navigation drop-down menu in the Administration console: Disaster Recovery System, Cisco Unified Serviceability, and Cisco Unified OS Administration interfaces.

- The CLI for the Controller.

The System Administrator has no access to functions in the Unified Intelligence Center Reporting application.

Note   If you configure unique System Administrator credentials for Member nodes, use those credentials to access the CLI for those Member servers.

CHAPTER 3

Cisco Unified Customer Voice Portal Templates for Cisco Unified Intelligence Center

This chapter presents the Cisco-designed reports for Cisco Unified Customer Voice Portal (Unified CVP). These templates are available to you once you import them and set their data source.

To become familiar with developing additional custom reports, run one of the stock reports, perform a Save As operation, and modify the Save As report.

This chapter contains the following topics:

Application Summary Reports (15, Daily, and Weekly)

Unified CVP has three Application Summary Reports: Application Summary 15, Application Summary Daily, and Application Summary Weekly.

These reports are useful for Dominant Path Analysis. They display which elements of the voice applications are being run and the number of times they are run.

The three reports display the same data but aggregate the data for different time periods. See Summary / Aggregate Tables, on page 79

Fields in this report are populated from these tables:

- ApplicationSummary_15 Table, on page 81

- ApplicationSummary_Daily Table, on page 81

- ApplicationSummary_Weekly Table, on page 82

- ResultRef Table, on page 95

- ElementtypeRef Table, on page 91

Fields in this report:

- **Date and Time**

  The date and time when the period of data collection began. From the dbdatetime field of the appropriate Application Summary table - 15, Daily, or Weekly.

- **Application Name**

  The name of the voice application. From the AppName field of the appropriate Application Summary table - 15, Daily, or Weekly.

- **Source Application Name**

  The name of the source application that transferred the element. This field typically shows *null*. From The SourceAppName field of the appropriate Application Summary table - 15, Daily, or Weekly.

- **Element Name**

  The name of the element. From the ElementName field of of the appropriate Application Summary table - 15, Daily, or Weekly.

- **Element Type**

  The element type. From the ElementTypeID field of the appropriate Application Summary table - 15, Daily, or Weekly.

- **Exit State**

  The exit state of the element. From the ExitState field of the appropriate Application Summary table - 15, Daily, or Weekly.

- **Result**

  Indicates how an element ended. From result, where ResultID = ResultRef.resulttypeID.

- **Count**

  The number of calls to run a particular application. From the Count field of the appropriate Application Summary table - 15, Daily, or Weekly.

- **Average Elapsed Time**

  The average number of seconds a call was in a particular element. Calculated by subtracting the start time from the end time over all the elements within the grouping.

Call Report

The Call report shows the first 500 calls, starting at the date/time you select in the Basic Filter.

For this report, you can take advantage of the Absolute Date feature of the Basic Filter and filter to a narrow range, for example, to a one-minute time interval:

Fields in this report are populated from these tables:

- Call Table
- CallTypeRef Table
- SubSystemTypeRef Table

Fields in this report:

- **Call Guid**

  The Global Unique ID (GUID) of a call. From Call.CallGuid.

- **ANI**

  The Automatic Number Identification (ANI) of the caller sent by the telephony provider.

  ANI does not need to be configured, and can therefore be a null field. From Call.ANI.

- **DNIS**

  The Dialed Number Identification Service (DNIS) sent by the telephony provider. From Call.DNIS.

- **UID**

  The external User Identifier (UID) of the originating caller, sent by telephony provider. From Call.UID.

- **UUI**

  The User-To-User Information (UUI) of the originating caller, sent by telephony provider. From Call.UUI.

- **IIDigits**

  The ANI II Digits of the originating caller, sent by the telephony provider. From Call.IIDigits.

- **Sub Subsystem Type**

  The type of Unified CVP Service used for the call, such as SIP, IVR, VXML, where Call.SubSystemTypeID = SubsystemTypeRef.SubsystemTypeRefID.

  See SubSystemTypeRef Table for the list of sybsystem types.

- **Call Type**

  The type of call, where Call.CallTypeID = CallTypeRef.CallTypeID. See CallTypeRef Table for the list of call types.

- **Call Started**

  The time when the call started. From Call.StartDateTime.

- **Call Ended**

  The time when the call ended. From Call.EndDateTime.

- **Time Persisted to Database**

  This is the timestamp when data is written to the database. If the reporting server is in Boston, and the Callserver is in San Diego, then calls placed in San Diego at 1 PM have a Coordinated Universal Time (UTC) startdatetime (+ 7 hours), and the dbdatetime for this call is 4 PM (Boston time).

- **Time Zone Offset from UTC (Minutes)**

  The offset in minutes of the local timezone from the UTC timezone. From Call.LocalTimeZoneOffset.

  If the reporting server is in Boston, and the Callserver is in San Diego, then calls placed in San Diego at 1 PM have a UTC startdatetime (+ 7 hours), and the dbdatetime for this call is 4 PM (Boston time). The timezone offset is 240 minutes.

- **Applications Visited**

The number of applications visited during the life of the call. From Call.NumAppVisited.

- **Errors**

  The number of errors that occurred during the call. From Call.NumError.

- **On Hold**

  The number of times the call was on hold due to unavailable ports. From Call.NumOnHold.

- **Opted Out**

  This will always show 0. From Call.NumOptOut, deprecated field.

- **Timed Out**

  The number of times the call timed out because it exceeded a processing time. From Call.NumTimedOut.

- **Transferred**

  The number of times the call was transferred out to an agent or to a Voice Response Unit (VRU) leg. The VRU leg is the leg that talks to VoiceXML Gateways.

Call Detail Report

This report displays robust detail for calls according to the filters you set. This report is based on a Stored Procedure and uses the *Filter Page for Stored Procedure and Anonymous Block Reports*.

Note that this is a wide report. You might need to set the printer to Landscape mode, use legal-sized paper, or Export to Excel.

Fields in this report include data from these tables:

- Call Table
- CallEvent Table
- CallICMInfo Table
- VXMLElement Table
- VXMLSession Table
- CallTypeRef Table
- CauseRef Table
- EventTypeRef Table
- ResultRef Table
- SubSystemTypeRef Table
- TransferTypeRef Table

Fields in this report are:

- **Call Start Date**

  The time when the call started. From Call.StartDateTime.

• **End Date and Time**

The time when the call ended. From Call.EndDateTime.

• **Local Timezone Offset**

The offset in minutes of the local timezone from UTC timezone. From Call.LocalTimeZoneOffset.

If the reporting server is in Boston, and the Callserver is in San Diego, then calls placed in San Diego at 1 PM will have a UTC startdatetime (+ 7 hours), and the dbdatetime for this call will be 4 PM (Boston time). The timezone offset is 240 minutes.

• **Call Guid**

The Global Unique ID (GUID) of a call. From Call.CallGuid.

• **Call Type**

The type of call, where Call.CallTypeID = CallTypeRef.CallTypeID.

• **ANI**

The Automatic Number Identification (ANI) of the caller sent by the telephony provider. From Call.CallGuid.

ANI does not need to be configured, and can therefore be a null field. From Call.ANI.

• **DNIS**

The Dialed Number Identification Service (DNIS) sent by the telephony provider. From Call.DNIS.

• **IIDigits**

The ANI II Digits of the originating caller, sent by the telephony provider. From Call.IIDigits.

• **UID**

The external User Identifier (UID) of the originating caller, sent by telephony provider. From Call.UID.

• **UUI**

The User-To-User Information (UUI) of the originating caller, sent by telephony provider. From Call.UUI.

• **Number of Applications Visited**

The number of applications visited during the life of the call. From Call.NumAppVisited.

• **Number of Interactions**

The number of applications visited during the life of the call. From VXMLElement.NumberofInteractions.

• **Number of Errors**

The number of errors that occurred during the call. From Call.NumError.

• **Number of Times On Hold**

The number of times the call was on hold due to unavailable ports. From Call.NumOnHold.

• **Number of Timeouts**

The number of times the call timed out because it exceeded a processing time. From Call.NumTimedOut.

• **Total Number of Transfers**

The number of times the call was transferred out to an agent or to a VRU leg. (*VRU* leg is the leg that talks to VoiceXML Gateways.) From Call.TotalTransfer.

- **Event Date and Time**

  The date and time of the event. From CallEvent.EventDateTime.

- **Event Type**

  The name of the event type, from Callback.eventtypeID, where Callback.eventtypeID = EventTypeRef.eventtypeid.

- **Subsystem**

  The Subsystem Name. From SubsystemTypeRef.Subsystem.

- **Subsystem Type ID**

  The ID of the Unified CVP Service used for the call, such as "2" for VXML. From SubsystemTypeRef.SubsystemTypeRefID.

- **CallLegId**

  This is an ID assigned by the service. From CallEvent.CallLegID.

- **Cause**

  The reason that the call event was generated. From CallEvent.CauseID.

- **Result**

  Indicates how an element ended. From result, where ResultID = ResultRef.resulttypeID.

- **Media Filename**

  This field is always *null* in this release. From CallEvent.MediaFileName.

- **Transfer Label**

  The destination to which Unified CVP transfers the call. From CallEvent.TransferLabel.

- **Transfer Type**

  The unique id of the transfer type. From CallEvent.TransferTypeID.

- **Messagebus**

  The name of the Call Server (its message adapter name) with which the Call Event is associated. From CallEvent.MessageBusName.

- **RouterCallKey**

  The Unified ICM router call key. From CallICMInfo.RouterCallKey.

- **RouterCallKeyDay**

  The Unified ICM router call key day. From CallICMInfo.RouterCallKeyDay.

- **RouterCallKeySeqenceNumber**

  The Unified ICM router call key sequence number. From CallICMInfo.RouterCallKeySequenceNumber.

- **Application Name**

The name of the VXML application. From VXMLSession.AppName.

- **Enter Date and Time**

  The date and time when the element was entered. From VXMLElement.EnterDateTime.

- **Exit Date and Time**

  The date and time when the element was exited. From VXML.ExitDateTime.

- **Duration**

  The length of the session. From VXMLSession.Duration.

- **Element Name**

  The name of the element. From VXMLElement.ElementName.

- **Element Type**

  The type of element. From VXMLElement.ElementTypeID.

- **Exit State**

  The exit state of the element. From VXMLElement.ExitState.

- **Session ID**

  The unique id of a VXML application session. From VXMLSession.SessionID.

- **Session Name**

  The name of the session assigned by VXML Server. From VXMLSession.SessionName.

- **Source Application Name**

  The name of the application that transferred to this one. From VXMLSession.SourceAppName.

- **VXML Cause**

  The reason that the application visit ended. From VXMLSession.CauseID.

- **VXML End Date and Time**

  The end date and time of the session. From VXMLSession.EndDateTime.

- **VXML Event Type**

  The mechanism used to end the application visit. From VXMLSession.EventTypeID.

- **VXML Start Date and Time**

  Date and time when session began. From VXMLSession.StartDateTime.

Call Traffic Reports (15, Daily, and Weekly)

Unified CVP has three Call Traffic Reports: Call Traffic 15, Call Traffic Daily, and Call Traffic Weekly.

These reports indicate the CallServer/VXML Server load during the course of the day. Call center administrators can monitor peak call volume times and monitor load levels on various call servers and VXML servers.

The three reports display the same data and aggregate the data for different time periods. See

Fields in this report are populated from these tables:

-

-

-

-

-

Fields in this report:

- **Date and Time**

  The date and time of the call, from the start of the increment. From the datetime field of the appropriate Call Traffic table (15, Daily, or Weekly).

- **SubSystem Type**

  The type of Unified CVP Service used for the call, such as SIP, IVR,VXML, where the Call Summary table SubSystemTypeID = SubsystemTypeRef.SubsystemTypeRefID.

  See for the list of subsystem types.

- **Call Type**

  The type of call, where the Call Summary Table CallTypeID = CallTypeRef.CallTypeID. See for the list of call types.

- **Number of Calls**

  Total number of calls in this period. From the NumCalls field of the appropriate Call Traffic table (15, Daily, or Weekly).

- **Average Call Length**

  The average call length. From the AvgCallLength field of the appropriate Call Traffic table (15, Daily, or Weekly).

- **Applications Visited**

  The total number of applications visited in this period. From the TotalAppVisited field of the appropriate Call Traffic table (15, Daily, or Weekly).

- **Errors**

  The total number of errors in this period. From the TotalError field of the appropriate Call Traffic table (15, Daily, or Weekly).

- **On Hold**

  The total number of times that calls were placed on hold in this period. From the TotalOnHold field of the appropriate Call Traffic table (15, Daily, or Weekly).

- **Opted Out**

This field is deprecated. From the TotalOptOut field of the appropriate Call Traffic table (15, Daily, or Weekly).

- **Timed Out**

  The total number of calls that timed out in this period. From the TotalTimeOut field of the appropriate Call Traffic table (15, Daily, or Weekly).

- **Transfers**

  The total number of calls were transferred in this period. From the TotalTransfer field of the appropriate Call Traffic table (15, Daily, or Weekly).

Call Traffic Charts

All three Call Traffic reports have chart views. To display the report data in the chart view, click the View drop-down and select Columns.

Call Traffic charts are columns that show the total Applications Visited, Errors, Number of Calls, On Hold calls, Timed Out calls, and Transfers for the 15 minute, Daily, or Weekly time interval.

Current and Historical Callback Reports

Unified CVP has two callback reports: Pending Callbacks and Historical Callbacks.

They both pull data from the CallBack Table.

Note Callbacks reports must use the `callback` database as the data source or an error occurs when you attempt to access the report. The error message received is *"Import could not be completed: Query validation failed against the selected data source."*

The Pending Callbacks report takes data from calls that have occurred in the past 30 minutes. The Historical Callbacks report shows calls that occurred prior to 30 minutes ago.

The CallBack Table is used to generate the report.

Fields in this report are:

- **ANI**

  This field specifies the Automatic Number Identification (ANI) for the call.

- **Callguid**

  This field specifies the call's unique identifier.

- **Cause**

  This field specifies the cause of the event (no answer, no response etc).

- **CVP estimated wait time**

  This specifies the estimated wait time by CVP.

- **Database date and time**

  This field specifies the name of the queue.

- **Date and Time**

  This field specifies the database update date and time.

- **Event date and time**

  This field specifies the several events that are generated during a Courtesy Call Back (CCB) call. It's the time of latest event.

- **Event type**

  This field specifies the type of the latest event.

- **Gateway**

  The identifier for the gateway, can be an IP address or a string identifier.

- **ICM estimated wait time**

  This field specifies the estimated wait time for ICM to start sending to CVP (at the time call comes into ICM scripts for the first time).

- **Location**

  This field specifies the location that is assigned to the ingress gate way.

- **ICM estimated wait time**

  This field specifies the estimated wait time for ICM to start sending to CVP (at the time call comes into ICM scripts for the first time).

- **Number of attempts**

  This field specifies the number of callback retries that have been made so far (in case the callback was not accepted earlier).

- **Queue enter time**

  This field specifies the time at which the call entered the queue.

- **Queue**

  This field specifies the name of the queue.

- **Queue leave time**

  This field specifies the time at which the call left the queue.

- **Queue status**

  This field specifies the current call back status of the call.

- **Recording URL**

  This field specifies the URL of the media file for caller's name.

- **Validation Status**

  This field specifies whether each call that gets into reporting server gets validated for CCB and the result is captured in a bit mask.

Trunk Group Utilization Report

This report shows a summary of trunk group utilization. It is a grid (a tabular display).

This report is populated by data from the Usage Table, the Resource Table, and the Device Table.

Fields in this report are:

- **Date and Time**

  The date for the event. From Usage.eventdatetime.

- **Device**

  The IP address of the device. From Usage.DeviceID, where Usage.DeviceID = Device.DeviceID.

- **Resource**

  The unique identifier of the resource being measured. From Usage.ResourceID, where Usage.ResourceID = Resource.ResourceID. See the Resource Table for the resource descriptors.

- **Average Resource Used**

  The average resource used, from Usage.ResourceUsed.

- **Max Threshold Reached**

  Was the maximum threshold reached (Yes | No), from Usage.ThresholdReached.

- **Max Resource Used**

  The maximum resource used, from Usage.ResourceMax.

CHAPTER **4**

Database Schema

About Database Schema

The Cisco Unified Customer Voice Portal (United CVP) reporting server hosts an IBM Informix Dynamic Server (IDS) database, which stores reporting data in a defined database schema. Customers who choose to deploy Cisco Unified Intelligence Center (Unified Intelligence Center) as their reporting platform must add the Informix database as a data source in Unified Intelligence Center.

The schema is fully published so that customers can develop custom reports. Customers may not, however, extend the schema for their own purposes.

The schema provides Unified CVP customers with the ability to:

- Establish database connectivity with Unified Intelligence Center and to import and run the Unified CVP templates with Unified Intelligence Center.

- Establish database connectivity with other commercial off-the-shelf reporting and analytics engines and then build custom reports against the Unified CVP database schema.

Note Your support provider cannot assist you with custom reports or with commercial (non-Cisco) reporting products.

The following diagram indicates a common set of incoming and outgoing entry and exit states for a call to a self-service application.

Figure 3: Call Flow
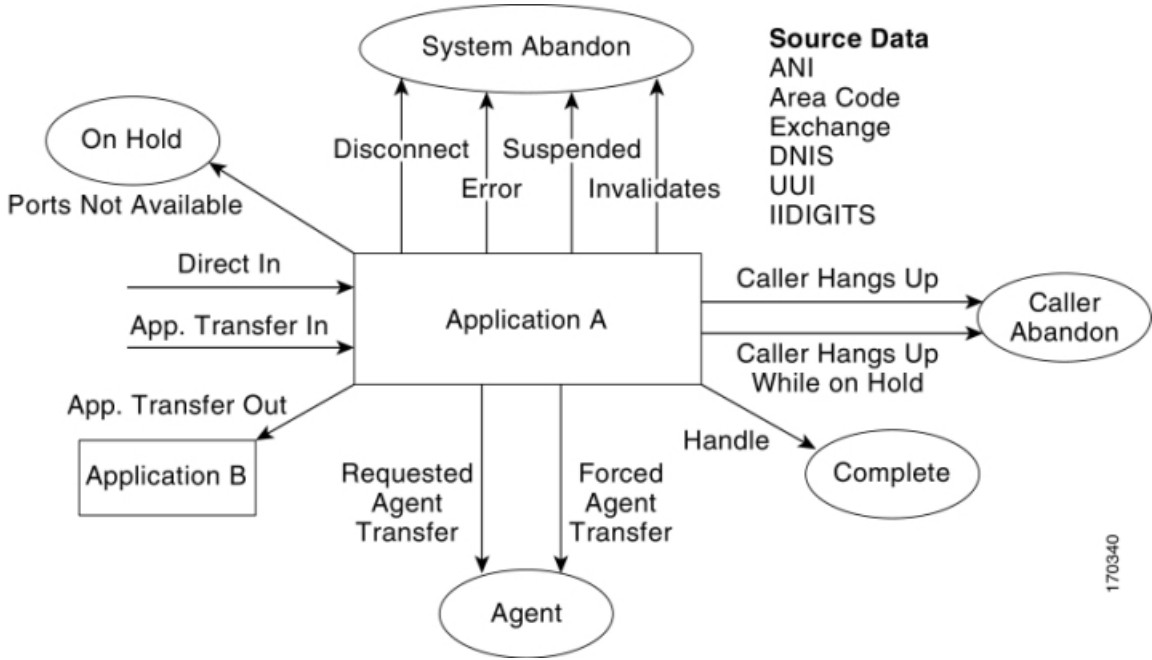
Note When basic video is transferred to an audio-only agent, the call remains classified as basic video accepted.

Data Model Diagram

The following entity-relationship diagrams depict the Unified CVP database schema.

Figure 4: Call Tables
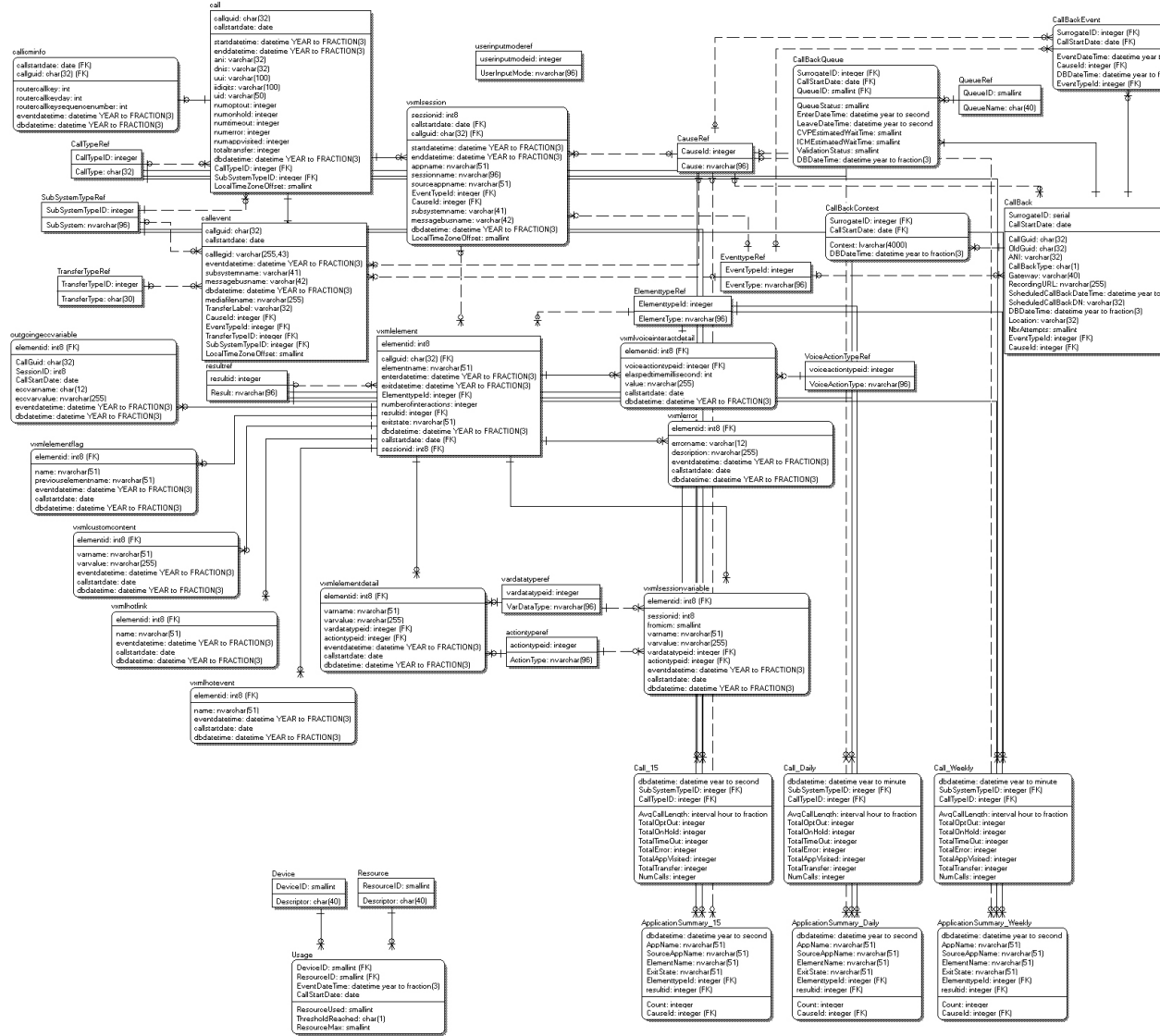
Figure 5: Callback Tables
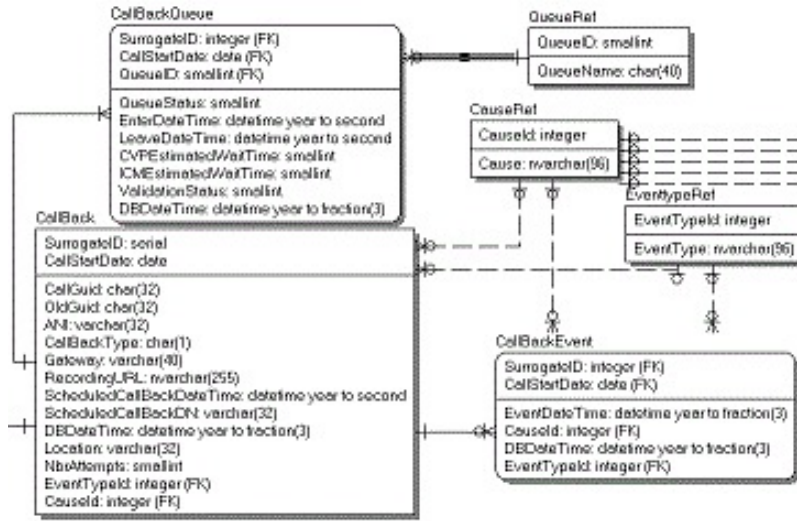
Figure 6: Summary Tables
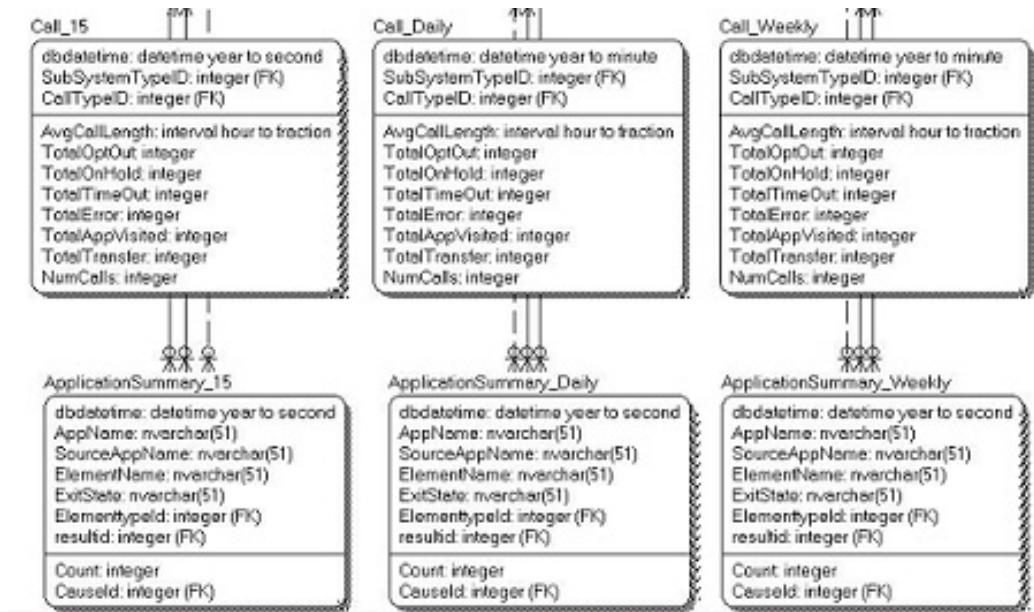
Figure 7: Trunk Group Utilization Tables
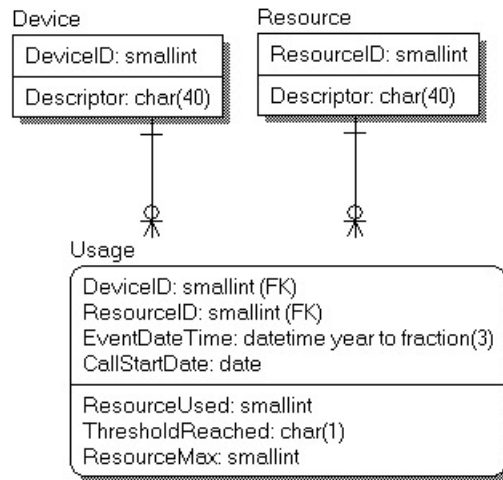
Keys

The documentation for the reporting schema lists fields as PK, FK, A, or No.

Fields are designated in this document as Primary Key (PK), Foreign Key (FK), or Alternate Key (AK) for informational purposes only. For performance reasons, the schema does not enforce Primary, Foreign, or Alternate keys. When the Index column for a field shows FK or AK, it means that a field can be looked up to return a text explanation of the code in another table.

Primary and Alternate Keys are in fact supported by an index. Major Foreign Keys (CallGUID, SessionID, ElementID) have a supporting index. Foreign Keys which refer to lookup tables are not supported by an index.

Unified Customer Voice Portal Reporting Data Model

The following section provides information on the following topics:

DateTime Columns

Most major tables have three columns to assist in managing the reporting server itself.

- **CallStartDate** - This column is used for partitioning and purging data.

This is the date and time the call started and is meant to ensure that detail data, which may cross a date boundary, are tied to the original call and can all be stored and removed together.

- **EventDateTime** - This is the date and time that the recorded event transpired. This is recorded in UTC time.

  The Call Table table has two EventDateTime fields, recorded as *StartDateTime* and *EndDateTime.*

- **DBDateTime** - This is the date and time that the recorded event was written to the database. It is meant to contrast with the EventDateTime. A marked difference between these values indicates a delay in the data arriving at the reporting server. This delay should either be allowed for or investigated.

Informix Dates and Times

The Informix engine that hosts the Unified CVP reporting database supports three concepts of time:

- Dates
- DateTimes
- Intervals

Dates

A date (for example CallStartDate) has no time element to it. It is specified between single quotes as 'MM/DD/YYYY'.

```
SELECT count(*)
  FROM Call
 WHERE CallStartDate='05/31/2012';
```

This date format can be modified to suit the locale with the DBDATE environment variable: in this case DBDATE=MDY4/, or Month/Day/Year(4) with a forward slash separator. These can be arranged in any order (DMY4-, or DMY2/ or Y4MD/) by modifying the DBDate enumeration variable.

Date also supports: key words such as 'TODAY' and date arithmetic.

For example, this returns a count of calls received yesterday:

```
SELECT count(*)
  FROM Call
 WHERE CallStartDate=TODAY-1;
```

Functions such as YEAR(), MONTH() and WEEKDAY().

```
SELECT count(*)
  From Call
 WHERE WEEKDAY(CallStartDate)=1
```

Note   Days of the week are numbered from 0 through 6 where 0 is Sunday and 6 is Saturday.

DateTimes

DateTimes include a time component and use the ANSI standard: 'YYYY-MM-DD HH:MM:SS.FFF' where FFF are fractions of seconds. For example, this returns a count of calls received in a given 48 hours:

```
SELECT count(*)
  FROM Call
 WHERE Call.StartDateTime between '2009-05-01 00:00:00' AND '2009-05-3
       23:59:59';
```

These support the same YEAR(), MONTH() and WEEKDAY() functions as the Date datatype. The Current date and time is specified as 'CURRENT YEAR TO SECOND' and also supports date arithmetic.

```
SELECT count(*)
  FROM Call
 WHERE Call.StartDateTime > CURRENT YEAR TO SECOND - 2 UNITS DAY;
```

Unified CVP DateTimes are all recorded as UTC time, with the exception of *dbdatetime* which is recorded as a local time. Localtimezoneoffset is a column in the Call table that contains the number of minutes offset from UTC to derive the Local Time. This can be used as an interval. (In the example below, localtimezoneoffset is -240 minutes).

```
select first 10 enddatetime, enddatetime + localtimezoneoffset units
minute as LocalTime from call;

enddatetime             localtime

2010-02-09 15:03:54.453 2010-02-09 11:03:54.453
2010-02-09 15:03:54.453 2010-02-09 11:03:54.453
2010-02-09 15:03:54.469 2010-02-09 11:03:54.469
2010-02-09 15:01:23.125 2010-02-09 11:01:23.125
2010-02-09 15:03:54.469 2010-02-09 11:03:54.469
2010-02-09 15:01:23.141 2010-02-09 11:01:23.141
2010-02-09 15:03:54.500 2010-02-09 11:03:54.500
2010-02-09 15:01:23.156 2010-02-09 11:01:23.156
2010-02-09 15:01:23.156 2010-02-09 11:01:23.156
2010-02-09 15:01:23.156 2010-02-09 11:01:23.156
```

An aggregation function lastperiod(datetime, *Period*) is supported. *Period* can be: 15, 30, 60, DD, WW, or MM. This will convert the datetime into the date and time at which the current period started. Hence:

Lastperiod(2009-10-14 12:46:56,*15*) returns *2009-10-14 12:45:00*

Lastperiod(2009-10-14 12:46:56, *30*) returns *2009-10-14 12:30:00*

Lastperiod(2009-10-14 12:46:56, *60*) returns *2009-10-14 12:00:00*

Lastperiod(2009-10-14 12:46:56, *DD*) returns *2009-10-14 00:00:00*

Lastperiod(2009-10-14 12:46:56, *WW*) returns *2009-10-11 00:00:00 (Sunday)*

Lastperiod(2009-10-14 12:46:56, *MM*) returns *2009-10-1 00:00:00 (1st day of the month)*

Intervals

An Interval is a span of time and can be specified as *n UNITS period* where *period* can be:

- YEAR

- MONTH

- DAY

- HOUR

- MINUTE

- SECOND

A database query with an interval must be sent in the preceding format. When returned from the database, the interval will look like a datetime (YYYY-MM-DD HH:MM:SS.FFF). The components that are returned depend on the interval definition. It is unlikely that a DAY component will be returned from Unified CVP intervals; instead, expect a format like HH:MM:SS.FFF.

For a full discussion of Informix, refer to the Informix Guide to SQL: Reference Manual.

SIP Calls

SIP calls are recorded in the Call Table along with VXML calls.

They can be distinguished from VXML calls with the CallTypeID column. (Contains "4". Refer to the CallTypeRef Table, on page 88, where 4 is a SIP call.)

Events for these calls (such as start and end) are recorded in the CallEvent Table.

Sample Query and SIP Calls

Details for a SIP call could be retrieved using the following query:

```
SELECT Call.*, CallEvent.*
  FROM Call, CallEvent
 WHERE Call.CallGUID=CallEvent.CallGuid
   AND Call.CallGuid='CallGuid';
```

where CallGuid is replaced by the value of the CallGuid for which information is desired.

Trunk Utilization

Trunk utilization is a record of state messages from various devices linked to the reporting server and their current status. The frequency in which these messages are written is controlled by the IOS Gateway (Gateway Utilization). This data captures a point-in-time over time. It is laid out in a fact table (Usage Table) with three dimensions - Resource, Device, and Time.

Because time is not likely to be consistent across all devices, the Usage table has not been codified as an official dimension table, but rather as a date and time. Queries for usage should aggregate from this table.

Sample Queries, Trunk Utilization

Query for average CPU across all devices for the month of May:

```
SELECT avg(ResourceUsed)
  FROM Usage, Resource
 WHERE Resource.ResourceID=Usage.ResourceID
   AND Resource= 'CPU'
```

```
                     AND Usage.EventDateTime between '2009-05-01 00:00:00' AND '2009-05-31
23:59:59';
```

Note that BETWEEN is inclusive. This query can also be written as:

```
               AND Usage.EventDateTime >=  '2009-05-01 00:00:00' AND Usage.EventDateTime <=
  '2009-05-31
               23:59:59';
```

Query for a list of devices and a count of the number of times they exceeded a threshold during the month of May:

```
        SELECT Device, Resource, count(*)
          FROM Device, Resource, Usage
         WHERE Resource.ResourceID=Usage.ResourceID
           AND Device.DeviceID=Usage.DeviceID
           AND Usage.ThresholdReached= 'Y'
           AND month(Usage.EventDateTime) = 5
        GROUP BY Device, Resource;
```

Note the use of the Month() function in *AND month (Usage.EventDateTime) = 5* .

Cisco Unified Customer Voice Portal Database Tables

This section lists the Unified CVP tables that hold reporting data.

Tables are categorized as follows:

Call Tables

The following Call tables are described in this section:

Call Table

This table is the primary record of a call and contains the basic metrics for each call. It contains one record per call.

Any drill into a specific call should start here to obtain the proper CallGUID.

On occasion, messages are dropped, even for an otherwise successful call. In such cases, EndDateTime is set to the same value as StartDateTime. Thus, if a call appears to be of 0 duration, report writers will know to exclude such a call from consideration in cases where it would otherwise skew metrics.

Table 6: Call Table

Field	Type	Null	Index	Description
CallGUID	char(32) for new installations char(35) for upgrades	No	PK (Composite CallGUID,CallStartDate)	The global unique id of a call.
CallStartDate	date	No	PK (Composite CallGUID,CallStartDate)	The date of the call, for data purging purposes.
StartDateTime	datetime YEAR to FRACTION(3)	No	Yes	EventDateTime for the date and time a call was made.
EndDateTime	datetime YEAR to FRACTION(3)	Yes	Yes	EventDateTime for the date and time a call ended or disconnected.
ANI	varchar(32)	Yes	No	The ANI of the caller sent by telephony provider
DNIS	varchar(32)	No	No	The DNIS of a call sent by telephony provider.
UUI	varchar(100)	Yes	No	The UUI of the originating caller sent by telephony provider.
IIdigits	varchar(100)	Yes	No	The IIDIGITS of the originating caller sent by telephony provider
UID	varchar(50)	Yes	No	The external UID of the caller if the call is associated with a user.
Numoptout (deprecated)	int	No	No	The number of times that the call is opted out to an agent.
NumTimeOut	int	No	No	The number of times the call timed out.

Field	Type	Null	Index	Description
NumError	int	No	No	The number of errors that occurred during the call.
NumOnHold	int	No	No	The number on hold within a VXML application.
NumAppVisited	int	No	No	The number of applications visited during the life of the call.
TotalTransfer	int	No	No	The total number of times the call is transferred out. A transfer includes transfers to agents as well as a transfer to the VRU leg.
DBDateTime	datetime YEAR to FRACTION (3)	No	Yes	The date and time of the database operation (when the record was inserted).
CallTypeID	smallint _Formerly char(1)_	No	Non-Indexed FK	The type of call. See CallTypeRef Table.
SubSystemTypeID	int	No	Non-Indexed KX	The type of Unified CVP Service, such as SIP, IVR, VXML.
LocalTime ZoneOffset	smallint		No	The offset in minutes of the local timezone from UTC timezone. _Replaces LocalTimeZone._

CallEvent Table

This table tracks each event that occurs within a call.

This table is populated for SIP calls. VXML calls will be recorded in the analogous VXMLSession Table.

Table 7: CallEvent Table

Field	Type	Null	Index	Description
CallGUID	char(32) for new installations char(35) for upgrades	No	Indexed FK (Composite CallGUID, CallStartDate)	The global unique id of a call

Field	Type	Null	Index	Description
CallStartDate	date	No	FK (Composite CallGUID, CallStartDate)	The date of the call, for data purging purposes
CallLegID	varchar(255,43) *Formerly Varchar(43)*	Yes	No	A call id assigned by a Service
MessageBusName	varchar(42)	No	No	The name of the Call Server (its message adapter name) with which the event is associated
EventDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time of the event
EventTypeID	int	No	Non-Indexed FK	The mechanism used to generate the call event. See EventTypeRef Table.
CauseID	int	No	Non-Indexed FK	The reason that the call event was generated. See CauseRef Table.
DBDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time of the database operation (when the record was inserted).
TransferTypeID	integer	No	Non-Indexed FK	A unique id of the transfer type. See TransferTypeRef Table.
SubSystemTypeID	int	No	Non-Indexed FK	The type of the Service. See SubSystemTypeRef Table.
SubSystemName	varchar(41)	No	No	The name of the Service the event originated from
MediaFileName	nvarchar(255)	Yes	No	*This is always null.*
TransferLabel	varchar(32)	Yes	No	This is the destination to which CVP transfers the call. The label is received from ICM via the TEMPORARY_CONNECT or CONNECT message
LocalTimeZoneOffset	smallint	NULL	No	The offset in minutes of this the local timezone from UTC timezone.

CallICMInfo Table

This table contains information to associate a Unified CVP call to ICM. It stores the ICM Call RouteCallKey, RouterCallKeyDay and RouterCallSequenceNumber for a call.

The CallICMInfo table is populated when the call is on the switch leg. This table is populated by SIP or VXML subsystems.

Note Currently the system does not capture the VRU leg of the call; thus if you have a Capture element and multiple Termination Call Detail (TCD) records are cut, the RouterCallKeySequenceNumber will increment in Historical Data Server (HDS) but will not be captured in the Unified CVP database. This is a known limitation.

Refer to the *Configuration Guide for Cisco Unified Customer Voice Portal* for further explanation about using the ReqICMLabel element to pass data to a Unified ICME script.

Table 8: CallICMInfo Table

Field	Type	Null	Index	Description
CallGUID	char(32) for new installations char(35) for upgrades	No	Indexed FK (Composite CallGUID, CallStartDate)	The global unique id of a call
CallStartDate	date	No	FK (Composite CallGUID, CallStartDate)	The date of the call, for data purging purposes

Field	Type	Null	Index	Description
RouterCallKey	Integer	No	AK (Composite index RouterCallKey, RouterCallKey Day)	ICM Router CallKey - single value per call. This value does not increment if the call is transferred from switch leg to VRU leg or if the call is transferred to an agent. If the call is a consult or conference, then Unified CVP will see two different callguids for the same call in its database. • The first callguid is the incoming callguid when the call is established. • The second callguid is for an agent originated/consult call. The RouterCallKey and RouterCallKey Day will act as a binder/glue between the two callguids for that single call as these values will not change between the two legs of the call.

Field	Type	Null	Index	Description
RouterCallKeyDay	Integer	No	AK(Composite index RouterCallKey, RouterCallKey Day)	ICM RouterCallKeyDay Typically this number changes on the switch and VRU leg of a call. You will see 0 for the switch leg of the call and 1 for the VRU leg of the call. This number usually does not change for basic CVP calls, but will increment if customers are using the capture node in their ICM script or when there is a transfer to an agent on the switch leg. In this scenario, Unified CVP sends a new call to Cisco Unified Communications Manager (Unified CM). This comes back via JTAPI and is on a separate Peripheral Gateway (PG). As the new call shows on a separate PG, Unified ICM cuts a new TCD record when the call ends. The RouterCallKeySequenceNumber increments on that switch leg.
RouterCallKey SequenceNumber	int	Yes	Yes	ICM RouterCallKeySequenceNumber.
EventDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time of the event.
DBDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time of the database operation (when the record was inserted).

VXML Tables

The following VXML tables are described in this section:

The data for VXML treatment is much richer than that which is available for SIP calls. Events can be captured from VXML for anything that occurs inside of the VXML script. These calls start at the Call Table, on page 64 and are linked to the VXMLSession Table, on page 76 using the CallGUID column.

The VXMLSession is made up of a series of elements that are visited within the context of an application. Each element may have multiple ancillary attributes such as flags that can be set in an element. Values for these flags may be found in the VXMLElementFlag Table, on page 73 and are linked to using the ElementID.

VXMLElementFlags information for a call can be retrieved using the following query:

```
SELECT VXMLElementFlag.Name
  FROM Call, VXMLSession, VXMLElement, VXMLElementFlag
 WHERE Call.CallGuid= CallGuid
AND Call.CallGuid=VXMLSession.CallGuid
AND VXMLSession.SessionID=VXMLElement.SessionID
AND VXMLElement.ElementID=VXMLElementFlag.ElementID;
```

where CallGuid is replaced by the value of the CallGuid for which information is desired.

VXMLCustomContent Table

This table contains one record for each VXML custom event. This event occurs if a custom component programmatically calls the AddToLog method of the Session API. The event will also occur when an element whose configuration contains entries in the Add To Log table in the General tab is run.

Table 9: VXMLCustomContent Table

Field	Type	Null	Index	Description
ElementID	int8	No	Indexed FK (Composite index ElementID, CallStartDate)	The unique ID of a visited element.
CallStartDate	date	No	FK (Composite index ElementID, CallStartDate)	The date of the call, for data purging purposes.
VarName	nvarchar(51)	No	No	The name of the custom event variable.

Field	Type	Null	Index	Description
VarValue	nvarchar(255)	Yes	No	The value of the custom event variable.
EventDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time when the variable is changed.
DBDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time of the database operation (when the record was inserted). This is useful for debugging purposes to determine lags between when the event occurred versus when it was written to the database (for example, a long lag may indicate problems with the reporting server).

VXMLElement Table

This table contains one record for each VXML script element visited by a call. For example, if the same element is visited twice in an application script during a call, there will be two separate element records.

Table 10: VXMLElement Table

Field	Type	Null	Index	Description
ElementID	int8	No	PK (Composite ElementID, CallStartDate)	The unique id of a visited element.
CallStartDate	date	No	PK (Composite ElementID, CallStartDate)	The date of the call, for data purging purposes.
SessionID	int8	No	Indexed FK	The unique id of a VXML application session.
CallGUID	char(32) for new installations char(35) for upgrades	No	FK	The global unique id of a call.
ElementName	nvarchar(51)	No	No	The name of an element.
ElementTypeID	int	No	Non-Indexed FK	The type of element.
EnterDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time when the element was entered.

Field	Type	Null	Index	Description
ExitDateTime	datetime YEAR to FRACTION(3)	Yes	No	Date and time when the element was exited.
ExitState	nvarchar(51)	Yes	No	The exit state of the element.
NumberOfInteractions	int	Yes	No	The number of interactions while the user visited this element.
ResultID	int	Yes	Non-Indexed FK	Indicates how an element ended.
DBDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time of the database operation (when the record was inserted). This is useful for debugging purposes to determine lags between when the event occurred versus when it was written to the database (for example, a long lag may indicate problems with the reporting server).

VXMLElementDetail Table

This table contains one detail record for each script element variable. VarValue holds the String value of the variable and VarDataTypeID specifies the data type of the variable to which the String value can be converted.

Table 11: VXMLElementDetail Table

Field	Type	Null	Index	Description
ElementID	int8	No	Indexed FK (Composite index ElementID, CallStartDate)	The unique id of an element.
CallStartDate	date	No	Yes (Composite index ElementID, CallStartDate)	The date of the call, for data purging purposes.
VarName	nvarchar(51)	No	No	The name of the element variable.
VarValue	nvarchar(255)	Yes	No	The String value of the element variable.

Field	Type	Null	Index	Description
VarDataTypeID	int	No	Non-Indexed FK	The data type of the element variable, such as String, Integer, Boolean.
ActionTypeID	int	No	Non-Indexed FK	The type of action for an element that changes data.
EventDateTime	datetime YEAR to FRACTION(3)	No	Yes	Date and time when the variable was changed.
DBDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time of the database operation (when the record was inserted).

VXMLElementFlag Table

This table contains one record for each element in which a flag was activated. The Name field holds the name of the flag.

Table 12: VXMLElementFlag Table

Field	Type	Null	Index	Description
ElementID	int8	No	Indexed FK (Composite index ElementID, CallStartDate)	The unique id for the element in which the flag activated.
CallStartDate	date	No	Yes (Composite index ElementID, CallStartDate)	The date of the call, for data purging purposes.
EventDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time when the flag activated.
Name	nvarchar(51)	No	No	The flag name.
PreviousElementName	nvarchar(51)	Yes	No	The name of the previous application element.
DBDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time of the database operation (when the record was inserted).

VXMLError Table

This table contains VXML errors that occurred during the life of the VXML application session. The table contains one record for each element in which an error occurred. The ErrorName field holds the name of the error.

Table 13: VXMLError Table

Field	Type	Null	Index	Description
ElementID	int8	No	Indexed FK (Composite index ElementID, CallStartDate)	The unique id for the element in which the error occurs.
CallStartDate	date	No	FK (Composite index ElementID, CallStartDate)	The date of the call, for data purging purposes.
ErrorName	varchar(12)	No	No	Name of an error.
EventDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time when the error occurred.
Description	nvarchar(255)	No	No	The detailed error message.
DBDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time of the database operation (when the record was inserted). This is useful for debugging purposes to determine lags between when the event occurred versus when it was written to the database (for example, a long lag may indicate problems with the reporting server).

VXMLHotEvent Table

HotEvent is a global event that when caught, runs developer-specified actions. This table contains information (HotEvent name, HotEvent DateTime and the ElementID) about the HotEvent occurred in an element.

Table 14: VXMLHotEvent Table

Field	Type	Null	Index	Description
ElementID	int8	No	Indexed FK (Composite index ElementID, CallStartDate)	The unique id for the element in which the HotEvent occurred.
CallStartDate	date	No	Yes (Composite index ElementID, CallStartDate)	The date of the call, for data purging purposes.

Field	Type	Null	Index	Description
EventDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time when HotEvent occurred.
Name	nvarchar(51)	No	No	The name of the HotEvent.
DBDateTime	datetime YEAR to FRACTION(3)	No	No	The date and time of the database operation (when the record was inserted). This is useful for debugging purposes to determine lags between when the event occurred versus when it was written to the database (for example, a long lag may indicate problems with the reporting server).

VXMLHotLink Table

Hotlink is a globally accessible utterance key press that immediately brings the call to a specific part of the call flow or throws an event. This table contains information (HotLink name, HotLink DateTime and the ElementID) about the HotLink that occurred in an element.

Table 15: VXMLHotLink Table

Field	Type	Null	Index	Description
ElementID	int8	No	Indexed FK (Composite index ElementID, CallStartDate)	The unique id for the element in which the hotlink activated.
CallStartDate	date	No	FK (Composite index ElementID, CallStartDate),	The unique id for the element in which the HotLink activated.
EventDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date of the call, for data purging purposes
Name	nvarchar(51)	No	No	The name of the HotLink.

Field	Type	Null	Index	Description
DBDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time of the database operation (when the record was inserted). This is useful for debugging purposes to determine lags between when the event occurred versus when it was written to the database (for example, a long lag may indicate problems with the reporting server).

VXMLSession Table

This table contains one record for each application visited by a VXML call. For example, if a call has transferred from one application to another one, the call with the same CallGUID will have two session records.

SIP calls are recorded in the .

Table 16: VXMLSession Table

Field	Type	Null	Index	Description
SessionID	int8	No	PK (Composite SessionID, CallStartDate)	The unique ID of a VXML application session.
CallStartDate	date	No	PK (second field in PK and Composite indexes)	The date of the call, for data purging purposes.
SessionName	nvarchar(96)	No	No	The name of the session assigned by VXML Server.
CallGUID	char(32) for new installations char(35) for upgrades	No	Indexed FK (Composite index CallGUID, CallStartDate);	The global unique id of a call.
StartDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time when session starts.
AppName	nvarchar(51)	No	Yes	The name of the VXML application.
EventTypeID	int	No	Non-Indexed FK	The mechanism used to end the application visit.

Field	Type	Null	Index	Description
CauseID	int	No	Non-Indexed FK	The reason that the application visit ended.
EndDateTime	datetime YEAR to FRACTION(3)	Yes	No	The end date and time of the session.
SourceAppName	nvarchar(51)	Yes	No	The name of the application that transferred to this one.
SubSystemName	varchar(41)	No	No	The name of the VXML Service.
MessageBusName	varchar(42)	No	No	The name of the message bus that delivers the VXML data feed message.
DBDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time of the database operation (when the record was inserted). This is useful for debugging purposes to determine lags between when the event occurred versus when it was written to the database (for example, a long lag may indicate problems with the reporting server).
Duration	int	Null	No	The length of the session.
LocalTimeZoneOffset	smallint	No	No	The offset in minutes of this the local timezone from UTC timezone. *Replaces LocalTimeZone.*

VXMLSessionVariable Table

This table contains one record for each instance of a session variable. For example, if the same session variable was modified once in an application script during a call, there will be two separate records, one for its initial value when it was created and another for the updated value.

Table 17: VXMLSessionVariable Table

Field	Type	Null	Index	Description
ElementID	int8	No	Indexed FK (Composite index ElementID, CallStartDate)	The identifier of the element in which the session variable changes.
SessionID	int8	No	Indexed FK (Composite index ElementID, CallStartDate)	The unique ID of an IVR application session.
CallStartDate	date	No	Yes (second field in Composite indexes)	The date of the call, for data purging purposes.
VarName	nvarchar(51)	No	No	The name of the session variable that was exited.
VarValue	nvarchar(255)	Yes	No	The value of the session variable.
ActionTypeID	int	No	Non-Indexed FK	The type of action for a session variable that changes data.
EventDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time when the session variable changed.
VarDataTypeID	int	No	Non-Indexed FK	The data type of the session variable, such as Integer, String, Boolean.
DBDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time of the database operation (when the record was inserted). This is useful for debugging purposes to determine lags between when the event occurred versus when it was written to the database (for example, a long lag may indicate problems with the reporting server).

Field	Type	Null	Index	Description
FromICM	Boolean	No	No	Indicates whether this session variable change originated from Unified ICME; Informix stores these values as 1 or 0, but represents these values as "t" or "f"

VXMLVoiceInteractDetail Table

This table has one record for each Voice Interaction with the caller.

Table 18: VXMLVoiceInteractDetail Table

Field	Type	Null	Index	Description
ElementID	int8	No	Indexed FK (Composite index ElementID, CallStartDate)	The unique ID of a visited element.
CallStartDate	date	No	FK (Composite index ElementID, CallStartDate);	The date of the call, for data purging purposes.
ElaspedTimeMillisecond	int	No	No	The time since the last interaction.
VoiceActionTypeID	int	No	Non-Indexed FK	The type of interaction.
Value	nvarchar(255)	Yes	No	The value of interaction.
DBDateTime	datetime YEAR to FRACTION(3)	No	No	The date and time of the database operation (when the record was inserted).

Summary / Aggregate Tables

The Summary / Aggregate tables are described in the following section:

- ApplicationSummary_15 Table
- ApplicationSummary_Daily Table
- ApplicationSummary_Weekly Table
- ApplicationSummary_Monthly Table, on page 83
- Call_15 Table

- Call_Daily Table

- Call_Weekly Table

Unified CVP reporting server includes a summary process that aggregates data from the Call and VXMLElement tables into new summary tables. These six tables hold summary data on Call metrics and on elements visited in Unified CVP applications.

These metrics include:

- The datetime of the beginning of the summary period.

- The average call length for the calls in this period

- Various totals, including the total number of opt outs, timeouts, and on holds for the calls in this period; the total number of transfers; and the total number of applications visited for the calls in this period

Summary tables use a star schema. Each summary table has a collection of non-numeric attributes and one or more numeric attributes that can be aggregated according to their type. Adding or removing an attribute from a query in a report definition allows a drill up or drill down into the data presented.

For example: the Application Summary tables have the following non-numeric attributes: Dbdatetime | Appname | Sourceappname | Elementname | Elementtypeid | Resultid | Causeid | Exitstate.

The numeric data available to report on those dimensions are: Avg_elapsed and Count.

`Select Appname, avg(avg_elapsed), sum(count)` will yield the average elapsed time and number of occurrences for an application. Adding ElementName to the Select clause (`Select Appname, ElementName, avg(avg_elapsed), sum(count)`) will further elaborate on where time was spent within the application. This can be further qualified by checking for specific Results, Causes, or Exit states.

These summary tables are not pure fact tables in cases where the dimensions are not always ID columns which refer to dimension or lookup tables.

In an upgrade situation, the summary process will start aggregation at the earliest data date within the Call and VXMLelement tables. At most, once every 15 minutes, the summary process will aggregate one day's worth of data from historical records to avoid overtaxing the system by attempting to process too much data.

This means that in a single 24-hour period, the system can summarize 96 days of data at most.

- _Daily tables will be populated one day behind the _15 minute tables.

- _Weekly tables will be populated from _Daily tables once those have been fully populated for the week in question.

- _Monthly tables will be populated from _Weekly tables once those have been fully populated for the month in question.

Retention for summary tables is hardcoded to 60 days for 15 minute summaries, 18 months for daily summaries, 10 years for weekly data, and 40 years for monthly aggregation.

> • Take into consideration that it can take some time to collect aggregate-level data from the reporting server.
> • Summary tables are built in 15-minute increments using the local time of the reporting server. Latency of source data is not guaranteed. In the event of a failover situation, data may arrive hours after it was initially created. For this reason all summary time periods reflect the time that the source data arrived at the database, which will generally be close to the time that it was created.

ApplicationSummary_15 Table

The ApplicationSummary_15 table is a 15-minute summary of Application/element data, useful for Dominant Path analysis.

Table 19: ApplicationSummary_15 Table

Field	Type	Null	Index	Description
dbdatetime	datetime year to second	Yes	PK	The start of the time period for this row.
AppName	nvarchar(51)	Yes	PK	The name of the VXML application.
SourceAppName	nvarchar(51)	Yes	PK	The name of the application that transferred to this one.
ElementName	nvarchar(51)	No	PK	The name of the element.
ExitState	nvarchar(51)	Yes	PK	The exit state of the element.
ElementTypeID	integer	Yes	PK; FK	The unique ID of an element type.
ResultID	integer	Yes	PK, FK	The unique ID of a result
Count	integer	Yes	No	The number of occurrences for this time period.
CauseId	int	Yes	FK	The unique ID of a cause.
Avg_elapsed	int	yes	No	The average elapsed time for this element.

ApplicationSummary_Daily Table

The ApplicationSummary_Daily table provides a daily summary of Application/element data, useful for Dominant Path analysis.

Table 20: ApplicationSummary_Daily Table

Field	Type	Null	Index	Description
dbdatetime	datetime year to second	Yes	PK	The start of the time period for this row.
AppName	nvarchar(51)	Yes	PK	The name of the VXML application.
SourceAppName	nvarchar(51)	Yes	PK	The name of the application that transferred to this one.
ElementName	nvarchar(51)	Yes	PK	The name of the element.
ExitState	nvarchar(51)	Yes	PK	The exit state of the element.
ElementTypeID	integer	Yes	PK, FK	The unique ID of an element type.
ResultID	integer	Yes	PK, FK	The unique ID of a result.
Count	integer	Yes	No	The number of occurrences for this time period.
CauseId	int	Yes	FK	The unique ID of a cause.
Avg_elapsed	int	yes	No	The average elapsed time for this element.

ApplicationSummary_Weekly Table

A weekly summary of Application/element data, useful for Dominant Path analysis.

Field	Type	Null	Index	Description
dbdatetime	datetime year to second	Yes	PK	The start of the time period for this row
AppName	nvarchar(51)	Yes	PK	The name of the VXML application
SourceAppName	nvarchar(51)	Yes	PK	The name of the application that transferred to this one
ElementName	nvarchar(51)	Yes	PK	The name of the element
ExitState	nvarchar(51)	Yes	PK	The exit state of the element

Field	Type	Null	Index	Description
ElementTypeID	integer	Yes	PK, FK	The unique id of an element type
ResultID	integer	Yes	PK, FK	The unique id of a result
Count	integer	Yes	No	The number of occurrences for this time period
CauseId	int	Yes	FK	The unique id of a cause
Avg_elapsed	int	yes	No	The average elapsed time for this element

ApplicationSummary_Monthly Table

The ApplicationSummary_Monthly table displays a monthly summary of Application/element data, useful for Dominant Path analysis.

Table 21: ApplicationSummary_Monthly Table

Field	Type	Null	Index	Description
dbdatetime	datetime year to second	Yes	PK	The start of the time period for this row.
AppName	nvarchar(51)	Yes	PK	The name of the VXML application.
SourceAppName	nvarchar(51)	Yes	PK	The name of the application that transferred to this one.
ElementName	nvarchar(51)	Yes	PK	The name of the element
ExitState	nvarchar(51)	Yes	PK	The exit state of the element.
ElementTypeID	integer	Yes	PK, FK	The unique ID of an element type.
ResultID	integer	Yes	PK, FK	The unique ID of a result.
Count	integer	Yes	No	The number of occurrences for this time period.
CauseId	int	Yes	FK	The unique id of a cause.
Avg_elapsed	int	yes	No	The average elapsed time for this element.

Call_15 Table

The Call_15 table dispalys a15-minute summary of call activity by SubSystemType and CallType.

Table 22: Call_15 Table

Field	Type	Null	Index	Description
dbdatetime	datetime year to second	Yes	PK	time in 15-minute increments.
SubSystemTypeID	integer	Yes	PK, FK	The unique ID of a Service type.
CallTypeID	smallint	No	PK, FK	The unique ID of a Call type.
AvgCallLength	interval HOUR (3) to FRACTION (3)	Yes	No	The average call length for this period.
TotalOptOut	integer	Yes	No	The total number of Opt Outs in this period.
TotalOnHold	integer	Yes	No	The total number of Holds in this period.
TotalTimeOut	integer	Yes	No	The total number of Time Outs in this period.
TotalError	integer	Yes	No	The total number of errors in this period.
TotalAppVisited	integer	Yes	No	The total number of applications visited in this period.
TotalTransfer	integer	Yes	No	The total number of transfers in this period.
NumCalls	integer	Yes	No	The total number of calls in this period.

Call_Daily Table

The Call_Daily table displays a daily summary of call activity by SubSystemType and CallType

Table 23: Call_Daily Table

Field	Type	Null	Index	Description
dbdatetime	datetime year to minute	Yes	PK	The time in daily increments.

Field	Type	Null	Index	Description
SubSystemTypeID	integer	Yes	PK, FK	The unique ID of a Service type.
CallTypeID	smallint	Yes	PK, FK	The unique ID of a Call type.
AvgCallLength	interval HOUR (3) to FRACTION (3)	Yes	No	The average call length for this period.
TotalOptOut	integer	Yes	No	The total number of Opt Outs in this period.
TotalOnHold	integer	Yes	No	The total number of Holds in this period.
TotalTimeOut	integer	Yes	No	The total number of Time Outs in this period.
TotalError	integer	Yes	No	The total number of errors in this period.
TotalAppVisited	integer	Yes	No	The total number of applications visited in this period.
TotalTransfer	integer	Yes	No	The total number of transfers in this period.
NumCalls	integer	Yes	No	The total number of calls in this period.

Call_Weekly Table

The Call_Weekly table displays a weekly summary of call activity by SubSystemType and CallType.

Table 24: Call_Weekly Table

Field	Type	Null	Index	Description
dbdatetime	datetime year to minute	Yes	PK	The time in weekly increments.
SubSystemTypeID	integer	Yes	PK, FK	The unique ID of a Service type.
CallTypeID	small int	Yes	PK, FK	The unique ID of a Call type.
AvgCallLength	interval HOUR (3) to FRACTION (3)	Yes	No	The average call length for this period.

Field	Type	Null	Index	Description
TotalOptOut	integer	Yes	No	The total number of Opt Outs in this period.
TotalOnHold	integer	Yes	No	The total number of Holds in this period.
TotalTimeOut	integer	Yes	No	The total number of Time Outs in this period.
TotalError	integer	Yes	No	The total number of errors in this period.
TotalAppVisited	integer	Yes	No	The total number of applications visited in this period.
TotalTransfer	integer	Yes	No	The total number of transfers in this period.
NumCalls	integer	Yes	No	The total number of calls in this period.

Call_Monthly Table

The Call_Monthly table displays a monthly summary of call activity by SubSystemType and CallType.

Table 25: Call_Monthly Table

Field	Type	Null	Index	Description
dbdatetime	datetime year to minute	Yes	PK	The time in weekly increments.
SubSystemTypeID	integer	Yes	PK, FK	The unique ID of a Service type.
CallTypeID	small int	Yes	PK, FK	The unique ID of a Call type.
AvgCallLength	interval HOUR (3) to FRACTION (3)	Yes	No	The average call length for this period.
TotalOptOut	integer	Yes	No	The total number of Opt Outs in this period.
TotalOnHold	integer	Yes	No	The total number of Holds in this period.
TotalTimeOut	integer	Yes	No	The total number of Time Outs in this period.

Field	Type	Null	Index	Description
TotalError	integer	Yes	No	The total number of errors in this period.
TotalAppVisited	integer	Yes	No	The total number of applications visited in this period.
TotalTransfer	integer	Yes	No	The total number of transfers in this period.
NumCalls	integer	Yes	No	The total number of calls in this period.

Lookup and Reference Tables

The Lookup and Reference tables are discussed in the following sections:

- ActionTypeRef Table
- CallTypeRef Table
- CauseRef Table
- Device Table
- ElementtypeRef Table
- EventTypeRef Table
- OutgoingECCVariable Table
- QueueRef Table
- Resource Table
- ResultRef Table
- SubSystemTypeRef Table
- TransferTypeRef Table
- Usage Table
- UserInputModeRef Table
- VarDataTypeRef Table
- VoiceActionTypeRef Table

ActionTypeRef Table

This is a reference table that resolves an ActionTypeID to the text value for an element that changes data.

Table 26: ActionTypeRef Table

Field	Type	Null	Index	Description
ActionTypeID	int	No	PK	The unique id of an action type.
ActionType *Formerly Name*	nvarchar(96)	No	No	The name of the action type.

Table Values (ID, Action Name):

1, "Initialize"

2, "Update"

3, "Return"

CallTypeRef Table

This is a reference table that resolves CallTypeID to a text value.

Table 27: CallTypeRef Table

Field	Type	Null	Index	Description
CallTypeID	small int	No	PK	The unique ID of a call type reference.
CallType	char(32)	No	No	The Call Type.

Table Values (ID, CallType):

1, "Legacy Audio"

2, "Legacy Video"

4, "SIP"

5, "VRU"

6, "VXML"

7, "Basic Video"

8, "Full Video"

CauseRef Table

This table maps a CauseID to the text value for the cause.

Table 28: CauseRef Table

Field	Type	Null	Index	Description
CauseID	int	No	PK	The unique ID of a call event cause.
Cause	nvarchar(96)	No	No	The cause of the event. *Formerly Name*

Table Values (ID, Cause):

0 = "None"

1, "Normal Completion"

2, "Call Abandon"

3, "Call Transferred"

4, "New Transaction"

5, "Busy"

6, "No Answer"

7, "Maintenance"

8, "Net Congestion"

9, "Net Not Obtainable"

10, "Reorder Tone"

11, "Resources Not Available"

12, "Trunks Busy"

13, "Called Party Disconnected"

14, "Max Ports"

15, "Suspended"

16, "Time Out"

17, "Invalidated"

18, "Error"

19, "Video Answered"

20, "Post Call Answer"

21, "Invalid"

22, "Failure"

23, "Audio Recording Start"

24, "Audio Recording Stop"

25, "No Response"

26, "Invalid Number"

27, "Connected"

28, "Caller Canceled"

29, "Whisper Start"

30, "Whisper Done"

31, "Whisper Setup Failed"

32, "Abandon In Whisper"

33, "Whisper Media Error"

1001, "Hang Up"

1002, "Network"

1003, "System"

1004, "Script Type"

1005, "Unknown UApp"

1006, "Script Name"

1007, "Config Param"

1008, "Misconfig Ecc"

1009, "Media File"

1010, "Semantic"

1011, "VXML Format"

1012, "VXML Element"

1013, "Variable Data"

1014, "No Var Data"

1015, "Format"

1016, "Entry Invalid"

1017, "No Entry"

1018, "Media Resource Video" [Unable to perform video-related request due to resource limitations]

1019, "Recording Failed"

1020, "Data Range"

1021, "Timed Out"

1022, "Called Hung Up" [Agent, VRU, or other endpoint stopped responding on a caller; that is, the caller did not stop responding first]

1023, "No Answer"

1024, "Busy"

1025, "Transfer"

1026, "Invalid Extn"

1027, "Hang Up Forced"

1028, "After Trans Estab"

1030, "Unsupported Language"

1031, "Media Resource ASR

1032, "Media Resource TTS"

1033, "General ASR TTS"

1034, "Unknown Error

1035, "Missing Configuration"

1036, "Duplicate Record"

1037, "Not in Queue"

1039, "Unknown Callguid"

1040, "CVP System Unavailable"

1041, "CVP App Error"

1042, "CVP App Hang Up"

1043, "Error CVP App Suspended"

1044, "Error CVP No Session Error"

1045, "Error CVP Bad Fetch"

1046, "No Streaming Media Resource TTS"

Device Table

The device for which this resource is measured. This is an IP Address.

Table 29: Device Table

Field	Type	Null	Index	Description
DeviceID	smallint	No	PK	Unique identifier of this device.
Descriptor	char(40)	Yes	No	The IP address of this device.

ElementtypeRef Table

This table maps an ElementTypeID to a text value for the VXML element type.

Table 30: ElementtypeRef Table

Field	Type	Null	Index	Description
ElementTypeID	int	No	PK	The unique id of an element type.

Field	Type	Null	Index	Description
ElementType	nvarchar(96)	No	Yes	The name of the element type . *Formerly Name*

Table Values (ID, ElementType):

0, Start"

1, "End"

2, "Subdialog_Start"

3, "Subdialog_Return"

4, "Decision"

5, "Action"

6, "Custom"

7, "HotLink"

8, "HotEvent"

9, "ElementFlag"

10, "Voice"

11, "VXMLInsert"

12, "ReqICMLabel"

13, "Genera"l

EventTypeRef Table

This is the table to map an EventID to the text value for its name (event type).

Field	Type	Null	Index	Description
EventTypeID	int	No	PK	The unique id of a call event type
EventType	nvarchar(96)	No	No	The name of the event type *Formerly Name*

Table Values (ID, EventType):

0, "New Call"

1, "Connect Failure"

2, "Busy"

3, "No Answer"

4, "Answer"

5, "Abandon"

6, "Disconnect"

7, "Hang Up"

8, "App Transfer"

9, "App Session Complete"

10, "Call Transfer"

11, "Run Script"

12, "Agent Recording"

13, "ICM Recording"

14, "Agent Video"

15, "ICM Video"

16, "Bridge Transfer"

17, "Blind Transfer"

18, "ReqICMLabel"

19, "Audio Recording"

20, "Callback Canceled"

21, "Callback Pending"

22, "Callback In Progress"

23, "Callback Tentative"

24, "Callback Complete"

25, "Callback Recover"

26, "Callback Created"

27, "Callback Dropped from queue"

28, "Callback Added to queue"

29, "Max allowed callbacks to this ANI exceeded"

OutgoingECCVariable Table

This table stores the ECC Variables that are returned from Unified CVP to an ICM script.

At present, this table is populated by the courtesy callback studio application element and when the ReqICMLabel element is used in a Call Studio script. Refer to the *CVP Administration and Configuration Guide* for further explanation about using the ReqICMLabel element to pass data to a Unified ICME script.

Table 31: OutgoingECCVariable Table

Field	Type	Null	Index	Description
CallGUID	char(32) for new installations char(35) for upgrades	No	Indexed FK (Composite index CallGUID, CallStartDate)	The global unique id of a call.
CallStartDate	date	No	FK (Composite CallGUID, CallStartDate)	The date of the call, for data purging purposes.
SessionID	int8	No	Yes	The identifier of the session in which the ECC variable changes.
ElementID	int8	No	Yes	The identifier of the element in which the ECC variable changes.
ECCVarName	char(12)	No	No	The name of session variable that was exited.
ECCVarValue	nvarchar(255)	No	No	The value of session variable.
EventDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time when the ECC variable changed.
DBDateTime	datetime YEAR to FRACTION(3)	No	Yes	The date and time of the database operation (when the record was inserted). This is useful for debugging purposes to determine lags between when the event occurred versus when it was written to the database (for example, a long lag may indicate problems with the reporting server).

QueueRef Table

QueueRef is a callback lookup table. This table maps QueueID to a text value for the queue in which a callback is waiting. The QueueName stores whatever you decide to call the queues.

Table 32: QueueRef Table

Field	Type	Null	Index	Description
QueueID	smallint	No	PK	The unique ID of a queue.
QueueName	char(40)	Yes	No	The name of the queue.

Resource Table

The resources that are measured include Memory, CPU, DSO and DSP.

Table 33: Resource Table

Field	Type	Null	Index	Description
ResourceID	smallint	No	PK	Unique Identifier.
Descriptor	char(40)	Yes	No	The name of the resource we are measuring (CPU, Memory, DSP, DS0, System).

ResultRef Table

This table maps a ResultID to a text value for a result.

Table 34: ResultRef Table

Field	Type	Null	Index	Description
ResultID	int	No	PK	The unique ID of a result.
Result	nvarchar(96)	No	No	The name of the element result. *Formerly Name*

Table Values (ID, Result):

1, "Normal"

2, "Invalidated"

3, "HotEvent"

4, "HotLink"

5, "Hang Up"

6, "Error"

7, "Transfer"

SubSystemTypeRef Table

This table maps a SubSystemTypeID to a Unified CVP Service type.

Table 35: SubSystemTypeRef Table

Field	Type	Null	Index	Description
SubSystemTypeID	int	No	PK	The unique ID of a Service type.
SubSystem	nvarchar(96)	No	No	The name of the Service type.

Table Values (ID, Name):

0, "SIP"

1, "IVR"

2, "VXML"

3, "OAMP" [Operate, Administer, Maintain, Provision = Operations Console]

4, "Controller"

5, "RPT"

6, "ICM"

7, "ORM" [Element with Unified CVP components that allows the Operations Console to manage the components]

8, "System"

TransferTypeRef Table

This is a reference table to resolve TransferTypeID to a text value.

Table 36: TransferTypeRef Table

Field	Type	Null	Index	Description
TransferTypeID	integer	No	PK	A unique ID of the transfer type.
TransferType	varchar(30)	Yes	No	The type of transfer performed.

Usage Table

This is a fact table of device/resource measurements.

Field	Type	Null	Index	Description
DeviceID	smallint	No	FK	Unique identifier of this device
ResourceID	smallint	No	FK	Unique Identifier
EventDateTime	datetime year to fraction(3)	No	PK	Date and time of this measurement
CallStartDate	date	No	PK	The date of this measurement for purge purposes
ResourceUsed	smallint	Yes	No	The amount of resouce used
ThresholdReached	char(1)	Yes	No	True/False. Was the maximum threshold for this resource reached?
ResourceMax	smallint	Yes	No	The amount of this resource available on this device

UserInputModeRef Table

This table maps a UserInputModeID to the name of the user input mode.

Table 37: UserInputModeRef Table

Field	Type	Null	Index	Description
UserInputModeID	int	No	PK	The unique ID of a user input mode.
UserInputMode	nvarchar(96)	No	No	The name of the user input mode. *Formerly Name*

Table Values (ID, Name):

1, "DTMF"

2, "Voice"

3, "DTMF Voice"

VarDataTypeRef Table

This table maps a VarDataTypeID to the data type of a variable.

Table 38: VarDataTypeRef Table

Field	Type	Null	Index	Description
VarDataTypeID	int	No	PK	The unique ID of a variable data type.
VarDataType	nvarchar(96)	No	No	The name of the variable data type. *Formerly Name*

Table Values (ID, Name):

0, "String"

1, "Int"

2, "Float"

3, "Boolean"

VoiceActionTypeRef Table

This table maps a VoiceActionTypeID to a text value.

Field	Type	Null	Index	Description
VoiceActionTypeID	int	No	PK	The unique ID of a VoiceActionTypeRef.
VoiceActionType	nvarchar(96)	No	No	The name of the call state. *Formerly Name*

Table Values (ID, Name):

1, "No Match"

2, "No Input"

3, "Audio Group"

4, "Input Mode"

5, "Utterance"

6, "Interpretation"

7, "Confidence"

Courtesy CallBack Tables

The following sections describe the Courtesy CallBack tables:

- CallBack Table
- CallBackEvent Table

- CallBackQueue Table

These tables support Courtesy Callback functionality.

Since this data is of an online-transaction-processing (OLTP) nature, it is retained in its own database, the callback database. When the caller registers a request for a callback, that request is stored in the CallBack Table.

A row is placed into the CallBackQueue Table for the call to manage timing and sequencing of calls.

Events that occur during the callback are registered in the CallBackQueue Table. This information can be retrieved using the following query:

```
SELECT Callback.*, CallBackEvent.*,
       CallBackQueue.*
  FROM Callback, CallBackEvent,
       CallBackQueue
 WHERE CallBack.oldGuid='testguid'
   AND CallBack.SurrogateID=CallBackEvent.SurrogateID
   AND CallBack.SurrogateID=CallBackQueue.SurrogateID;
```

Where CallGuid is replaced by the value of the CallGuid for which information is desired.

Query for number of callbacks currently pending:

```
SELECT count(*)
  FROM CallBack, EventTypeRef
 WHERE CallBack.EventTypeID=EventTypeRef.EventTypeID
   AND EventType in ('Callback Pending');
```

Query for a list of failed callbacks with telephone number and failure reason code:

```
SELECT *
  FROM CallBack, CauseRef, EventTypeRef
 WHERE CallBack.CauseID=CauseRef.CauseID
   AND CallBack.EventTypeID=EventTypeRef.EventTypeID
   AND EventType in ('Callback Canceled');
```

CallBack Table

The callback table is a view of two tables: Callback_Current and Callback_Historical. The two tables are identical; every 30 minutes, data for completed calls is pulled from Callback_Current and moved to Callback_Historical.

One row is generated in this table for each callback that is made.

Table 39: CallBack Table

Field	Type	Null	Index	Description
SurrogateID	serial	No	PK	A unique generated key to replace the CallGuid throughout the Callback schema.

Field	Type	Null	Index	Description
CallStartDate	date	No	PK	The date of the callback for data purging purposes.
OldGuid	char(32)	Yes	No	The Guid for the original scheduled callback. Used by the DB servlet to retrieve information about the old scheduled callback in order to create a new record for the pre-emptive callback.
ANI	varchar(32)	Yes	No	The number to which the callback will be placed.
CallBackType	char(1)	Yes	No	P = preemptive. S = scheduled
Gateway	varchar(40)	Yes	No	The identifier for the gateway. Can be an IP address or other string identifier.
RecordingURL	nvarchar(255)	Yes	No	The URL that points to a wav file of the recorded name of the caller.
ScheduledCallBack DateTime	datetime year to second	Yes	No	The Datetime (including timezone) for a scheduled callback. Not included for preemptive callbacks.
ScheduledCallBackDN	varchar(32)	Yes	No	The DN to which the scheduled callback will be placed. This will invoke an ICM script with this DN.
DBDateTime	datetime year to fraction(3)	Yes	No	The date and time of the database operation.
Location	varchar(32)	Yes	No	The location name assigned to a set of gateways. Used in scheduled callback to select applicable egress gateways for the callback.
NbrAttempts	smallint	Yes	No	The number of attempts made to call back.

Field	Type	Null	Index	Description
EventTypeId	integer	Yes	FK	The unique ID of a event type.
CauseId	integer	Yes	FK	The unique ID of a cause.

CallBackEvent Table

This table holds a record of each callback event that occurs for the call.

This table holds seven days worth of data. Purge daily.

Table 40: CallBackEvent Table

Field	Type	Null	Index	Description
SurrogateID	integer	No	PK, FK	A unique generated key to replace the CallGuid throughout the Callback schema.
CallStartDate	date	No	PK	The date of the callback for data purging purposes
EventDateTime	datetime year to fraction(3)	No	No	The date and time of the event
CauseId	integer	Yes	Yes	See Values below. CauseRef Table
DBDateTime	datetime year to fraction(3)	No	No	The date and time of the database operation.
EventTypeId	integer	No	FK	The unique ID of an event type. See ElementtypeRef Table.

Table Values (ID, CauseID)

0, "busy"

1, "noanswer"

2, "noresponse"

3, "invalid_number"

4, "connected"

5, "caller_canceled"

6, "trunksbusy"

7, "error"

CallBackQueue Table

This table holds data for the queue in which the call sits until its scheduled time or until a slot becomes available.

Table 41: CallBackQueue Table

Field	Type	Null	Index	Description
SurrogateID	integer (serial)	No	PK, FK	
CallStartDate	date	No	PK	The date of the callback for data purging purposes.
CallGuid	char(32) for new installations char(35) for upgrades	Yes	No	Unique ID for the call.
QueueID	smallint	No	FK	
QueueStatus	smallint	No	No	Status in queue: 0 = in_queue, 1 = not_in_queue 2 = Zombie
EnterDateTime	datetime year to second	No	No	The Datetime entered queue.
LeaveDateTime	datetime year to second	Yes	No	The Datetime left queue.
CVPEstimatedWaitTime	smallint	No	No	The CVP-calculated estimated wait time (in seconds) Since enterdatetime. This is generated during the insert, it will not be maintained.
ICMEstimatedWaitTime	smallint	No	No	Unified ICM-calculated estimated wait time (in seconds) Since enterdatetime. This is generated during the insert, it will not be maintained.
ValidationStatus	smallint	No	No	The bitmask result obtained from the Validation method. See sample code that follows this table.

Field	Type	Null	Index	Description
DBDateTime	datetime year to fraction(3)	No	No	The date and time of the database operation.

Validation Method sample code.

This is an example of a bitmask result obtained from the Validation method:

- TOD, Time of Day Error, meaning the callback was scheduled for a time of day when the queue is not open.

- EWT, Estimated Wait Time, indicates if the agent wait time for an agent is long enough to warrant a callback.

```
00000000 00000001 OK
00000000 00000010 ICM_NO_SCHEDULED_ALLOWED
00000000 00000100 ICM_NO_PREEMPTIVE_ALLOWED
00000000 00001000 NOT_IN_QUEUE
00000000 00010000 TOD
00000000 00100000 EWT
00000000 01000000 PROBE_FAILED_NO_RESPONSE
00000000 10000000 PROBE_FAILED_NO_CONFIG
00000001 00000000 EXCEED_CAPACITY_GW
00000010 00000000 EXCEED_CAPACITY_QUEUE
```

C H A P T E R 5

Database Management

Cisco Unified Customer Voice Portal (Unified CVP) provides access to database maintenance tasks such as database backups and data purges through the Operations Console.

Although the Reporting Service does not directly perform database administrative and maintenance activities such as backups or purges, familiarize yourself with the database management concepts discussed in this chapter.

✎

Note   Operations such as purge and changing user passwords are run through OAMP by the Administrator cvp_dbadmin.

Passwords

Passwords on the reporting server *must* be created and updated as part of the Unified CVP installation or by means of the Operations Console. Do not use any other means to create or update passwords.

Passwords for the Unified CVP Reporting Server are kept encrypted locally on the OAMP server and are set on the reporting server. If these passwords are not in synch (are not identical), the Operations Console will be unable to communicate with the reporting server, and restricted operations such as the purge will fail.

If you implement a password expiration policy, remember to use the Operations Console to change the Database Administrator, and Database Users passwords before the passwords expire to avoid the possibility of data loss or downtime.

✎

Note   Using the Operations Console to change passwords and to renew passwords before they expire ensures that all dependencies are synchronized.

⚠

Caution   Changing passwords outside of the Operations Console can result in a failed connection between the reporting server and the database.

Reporting passwords are subject to both the Unified CVP password policy *and* the password policy enforced by the operating system of the computer on which the reporting server resides and must meet the requirements of the more restrictive policy.

Database Users

Unified CVP defines three categories of database users: database administrator, application user, and reporting user.

Database Administrator

The `cvp_dbadmin` creates, updates, and owns the database.

This user can create and delete reporting users and perform database administrative activities, such as purge and backup.

This account should not be used to run the database or to run reports against the system.

✎

Note   If this administrator's password expires, then data insertion and purge will fail, which could result in data loss.

Application User

The Unified CVP JDBC uses `cvp_dbuser` to access the Informix database. This user has the rights to connect, insert, update, and delete records in the Unified CVP database. If this user's password expires, then data insertion and purge will fail, which could result in data loss.

The User ID and password for the application user is required to access the Cisco Unified Intelligence Center data source.

Reporting User

The Unified CVP OAMP has a UI page for creating Reporting users who have read-only database access to the Unified CVP Informix reporting database.

After the Active Directory configuration for these users is enabled in the Unified IC Administration Console, they can log in to Unified IC reporting with their AD credentials.

They have the basic "Login User" user role only, until the Unified IC Security Administrator assigns additional roles and privileges to them.

Data Categories and Data Retention

Using the Operations Console, users are able to select the time of day to run database purge and to set the number of days of data to be retained by data category. During schema creation, default data retention values are specified for each data category. Note that a high-level category, such as Call, cannot have a lower retention time than a dependent category, such as Call Event.

Increased database space availability, as documented in the *Solution Design Guide for Cisco Unified Contact Center Enterprise*.

Default Data Retention for Data Categories

The following data categories exist for Unified CVP. Note that a high level category, such as Call, cannot have a lower retention time than a dependent category, such as CallEvent. For each category, the default data retention times, in days, is given within parentheses.

level 1: `Call` (30)

level 2: `-Call Event` (30)

level 2: `-VoiceXML Session` (30)

level 3: `--VoiceXML Element` (15)

level 4: `---VoiceXML ECC Variable` (15)

level 4: `---VoiceXML Interact Detail` (15)

level 4: `---VoiceXML Session Variable` (15)

level 4: `---VoiceXML Element Detail` (15)

Database Purge

This section explains how to schedule purges, the difference between midday and nightly purges, and how to run emergency purges.

To allow for rapid space management, all Unified CVP Reporting Server data is kept in date-specific fragments. On a daily basis, new day fragments are created for incoming data. This allows the Unified CVP Reporting Server to quickly drop old data by dropping the disk fragment in which that data resides.

This means that new fragments must be created on a regular basis to ensure that they can be rapidly disposed of when their retention period expires. All of this is handled by the purge.

In a quiet environment, the purge can run in less than one second. In a busy environment, the purge might take considerably longer (15-20 minutes) while the purge navigates around running processes.

To allow for situations where the purge may be unable to run, space is allocated two days before it is needed. This allocation is triggered by the first purge to run after a date boundary has been crossed.

If the nightly purge is scheduled to run at 1:00 a.m., it will typically perform this task. If the nightly purge is scheduled to run at 11:00 p.m., then the next day will not occur until 11:00 a.m. the next morning, which may not be an optimal time to run the process.

For this reason, schedule the nightly purge to occur after midnight.

Schedule Purges

To run database purge from the Operations Console:

Procedure

Step 1	Select **Device Management** > **CVP Reporting Server**.
Step 2	Select a reporting server by clicking on the link in its name field or by clicking the radio button preceding it and then clicking **Edit**.
Step 3	At the Edit Reporting Server Configuration window, select the Database Administration menu in the toolbar, then select **Data Delete**.
Step 4	On the Reporting Server - Data Delete page, change the data retention time for each category of data.
Step 5	Select the hours and minutes to run the purge each day. This defines the time for the primary (nightly) purge and sets the Midday purge to run 12 hours later.
Step 6	Click **Save & Deploy**.

What to do next

See the *Administration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/en/US/products/sw/custcosw/ps1006/products_user_guide_list.html for information on categories of reporting data that can be purged and the default number of days to retain data before purging.

Nightly and Midday Purges

When you schedule a purge from the Operations Console, two jobs are scheduled on the reporting server - the nightly purge and the midday purge:

- The **nightly purge** job runs at the time (hour and minute) that you define in the Operations Console. The nightly purge performs a purge if necessary (as required by a data retention value, or for an emergency purge—see the following section on emergency purges), in addition to other tasks updating the database statistics.

  If a purge *is* required and performed, the statistics are updated after the purge.

  In addition, on Sundays, the nightly purge also copies the Informix log file to a backup folder, creates a new log file and deletes the old.

  The nightly purge should be scheduled after midnight.

- The **midday purge** is automatically scheduled to run 12 hours after the nightly purge. So, for example, if you schedule a purge at 2 a.m., then the nightly purge is run at 2 a.m. and the midday purge at 2 p.m.

Midday purge also serves as a backup for the nightly purge. If the nightly purge fails to allocate new fragments for new data, this will be taken care of by the midday purge. A midday purge is not system intensive in the same way that the nightly purge is.

In the event that data volume spikes during the day and an emergency purge is required, it will be handled at midday.

Emergency Purge

If the number of days of data that you chose to retain cannot be contained within the database, then the database initiates emergency purge. It purges the old data to create space for new data. Emergency purge is a critical safety mechanism for Unified CVP.

If used space exceeds the systems threshold, a Simple Network Management Protocol (SNMP) trap message notifies the user after the emergency purge is complete. The SNMP notification alerts the user of the loss of data and request that they reduce their retention days data settings.

Reduce the number of days of data retained so that emergency purge is not required. Also, you can reduce the data generated by using data filters (for VXML Server application detail data filtering).

Emergency Purge is triggered based on the database size and the free space value of the database. The **partitionparameters** table in the ciscoadmin database has the fields **dbsize** and **pctfree** which indicates the database size and percentage of the database size which is free respectively. For example, If the database size is 100 GB and free space is 17%, then the emergency purge starts when the database is 83% filled. It purges the oldest data until the required free space is achieved.

Guidelines for Purge

- **Data Granularity** - The CVP Reporting database houses records of calls handled by Cisco Unified Voice call servers. The amount of data captured for each call is managed by filters specified in the Operations Console. The granularity of data captured depends on these settings.

- **Data Retention** - As limited space is available to capture this data, the purge mechanism uses retention settings to govern how long data is retained.

  If there is insufficient space to retain data for the desired time frame, the oldest data is purged in one-day increments until there is sufficient space for the reporting server to remain operational.

  If more data is captured on a daily basis than can be stored, the purge mechanism will be unable to remove data because it operates only on a daily basis. If this is the case, consider installing a larger reporting server.

- **Database backup and purge** cannot run at the same time. Purge should be scheduled at least 30 minutes before a backup. These jobs, as well as on-demand backup, should be run at low call and reporting volume times. From the perspective of Unified CVP, database backups are optional, data purges are mandatory. However, from the perspective of the user, database backups should *not* be considered optional.

- **Reporting Server** - During a database purge operation, the reporting server disconnects from the database (though for no more than 10 minutes) and starts buffering messages in memory until the purge is finished. The same memory limitations apply as described in the section Reporting User.

- **Reporting users** may be disconnected from the database if they are holding locks that contend with purge. Notify reporting users not to run reports at this time.

- **Upgrades** - Turn off scheduled purge before performing an upgrade.

- **Windows Scheduled Tasks**- The database backup and purge maintenance tasks are created as Windows Scheduled Tasks, and can be viewed in the Scheduled Tasks window. (**Start** > **Programs** > **Accessories** > **System Tools** > **Scheduled Tasks**.) Periodically, you should check the Scheduled Tasks to ensure the Last Run Time was as expected and no status messages exist.

Note   In case of a reporting database purge or log purge failure, check whether the CVP DB is connected.

If the DB is not connected, stop all services. Restart the Call Server followed by the WSM server to restore the database purging.

Database Backup

Unified CVP lets users turn the scheduling of data backups on or off, and to run backups on demand. Backups are made to the reporting servers local file system. By default, scheduled backups are turned off.

Caution   Unified CVP backup scheduling is an optional feature. Backup is the user's responsibility. Data loss may occur if the backing up of files is not managed properly by the user.

Backup and restore of database from the same version and the same machine is supported.

If Unified CVP backup scheduling is turned on, the backup occurs once per day. Backups must be scheduled to run no sooner than 30 minutes after the scheduled purge job.

Users can run a backup on demand—as long as another backup, or a purge, is not already running. Database backups are performed and stored on the local machine. Due to space limitations, a maximum of two backups and a minimum of one backup are available on the local machine. Retaining two backup files is critical. If the system fails while writing a backup, and a restore is necessary, the older backup file is required for restore.

Follow these guidelines:

- Keep a given backup for at least two weeks.

- Check the integrity of the backup periodically.

- Run a backup before an upgrade.

Unified CVP uses the Informix backup utility ontape (for both backup and restore).

When a new backup launches—either scheduled, or on demand from the Operations Console—the new file is named `cvp_backup_data.gz`. The Unified CVP backup script copies the previous `cvp_backup_data.gz` backup file and renames it to `cvp_backup_data.old.gz`. This always leaves two backup files on the local system and makes it easy for Unified CVP administrators to script copy jobs to move the files. The backup script ensures that two backups cannot be launched at the same time.

Note   The backup script also ensures that a backup cannot be launched if a purge is underway, and vice versa.

☞

Important   You must manually or automatically (by creating an automated job) copy the `cvp_backup_data` files to a separate machine at a separate location. This will prevent accidental deletion of the files when the CVP machine fails or when CVP needs to be installed / uninstalled.

⚡

Warning   Only the `cvp_backup_data.old.gz` file can be copied. The `cvp_backup_data.gz` file cannot be copied. Attempting to copy the `cvp_backup_data.gz` file locks the file and prevent another backup from running.

Database backup updates the log file when the backup finishes. If the database server goes down during backup, the backup file gets corrupted.

To check if Informix is up and running and to validate the backup, run the `cvpverifybackup.bat` file located at `%CVP_Home%\bin\cvpverifybackup.bat`.

While running the script, you are prompted with following message:

```
Please put in Phys Tape 1.
Type <return> or 0 to end:
Press 0 and press Enter
```

✎

Note   Based on the size of the database, the prompt keeps changing as Tape 2, Tape 3 and so on.

This process takes a long time (based on the database size) to validate and the results are displayed on the console.

In Cisco Unified CVP, there is a supported script to perform a database restore.

Restoring a backup image is required when older data on a backup image needs to be recovered. It is also required when a machine is rebuilt after a hardware failure and you need to recover data.

✎

Note   Although it is possible to restore a backup image from one reporting server to another, such a restoration is not supported with the CVP restore process.

The restore process in Unified CVP is as follows:

- Stop the CallServer process (Reporting Server).

- Run the script: `%CVP_Home%\bin\cvprestore.bat`.

- Restart the CallServer process.

⚠

Caution   Using a third-party backup utility to back up the Informix database is ineffective and may be dangerous to the integrity of the reporting database. The only effective way to perform a reporting database backup is with the backup process provided by the OAMP interface.

For information on configuring backups, see the *Administration Guide for Cisco Unified Customer Voice Portal*.

Backup and Purge Retries

Occasionally, a backup or purge cannot run when scheduled. For example, if an on-demand backup is running when a purge is scheduled to run, the purge will be prevented from running.

Retries of scheduled backups or purges are performed according to the following rules.

| Note | There are no retries for an on-demand backup. |

- A scheduled backup retries every 10 minutes, for up to 4 hours.

- A purge retries every 10 minutes, for up to 6 hours.

- At the end of 4 hours (for a backup) or 6 hours (for a purge), if the operation has not succeeded, retries stop and an SNMP alert is sent.

- If both a backup and a purge are retrying simultaneously, there is no guarantee as to which operation will run first.

- If a lock (the mechanism preventing a backup or purge from running) is more than 12 hours old, the system clears it.

Database Recovery

Unified CVP database recovery returns the database to the state of the most recent complete backup. For example, if the user schedules a backup at 01:00 and restores the database at 23:00, the same day, the restored database is in the state it was in at 01:00.

During a database restore, the database will go offline for the duration of the restore operation.

| Note | Data loss occurs if the reporting server is turned off and the message bus exceeds its temporary persistence capabilities. |

| Caution | Before following a database restore, the following steps must be performed: |

1. Before the restore, disable scheduled tasks (backup, purge).

2. After the restore, re-enable scheduled tasks.

Failure and Restoration

If the reporting server fails, messages destined for the reporting server are buffered by the Call Server, in memory, up to 200,000 messages. After that limit is reached, all new message detail information is dropped.

If the database connection fails, the reporting server sends out an SNMP alert and starts persisting messages to a file, up to a user-specified limit. During this time the reporting server stays *In Service*. When 75% of the specified limit is reached, a warning is written to the log file. Once 100% of the limit is reached, an SNMP alert is sent out and the reporting server goes into *Partial Service*—any new messages may be dropped.

When the database connection comes back up, the reporting server goes into recovery mode and changes its state to Partial Service if it is not in that state already. It then starts reading messages from the file and committing them to the database. Depending on the size of the file, it may take a long time (sometimes hours) to commit all of the data to the database. Any new messages that come in during recovery will be buffered in memory. There is, however, a limit to the number of messages that the reporting server can buffer. This is true regardless of the mode or state it is in. When the number of buffered messages reaches 100,000, an SNMP alert is sent out to warn the user. At 200,000 another SNMP alert is sent out and all new messages detail information is dropped—keeping only basic data like call, call event, and session information. Also at 200,000, the reporting server changes its state to *Partial Service*, if it is not already in that state. After the total number of buffered messages reaches 300,000, another SNMP alert is sent out and all new messages are dropped from that point forward.

When the number of messages in memory drops back below 50,000, an SNMP alert is sent out stating that the queue size is back to usual, and the reporting server's state goes back to *In Service*.

If, on startup, a persistent file exists, the reporting server stays in *Partial Service* and goes into recovery mode as previously described.

During a database purge operation, the reporting server disconnects from the database and starts buffering messages in memory until the purge is done. The same memory limitations as previously described apply in this case as well.

⚠️

Caution   **When the reporting server is in *Partial Service*, there are no guarantees that new messages will be kept and committed to the database. As many as possible will be buffered in memory, but at some point they may be dropped either partially or fully.**

Reporting Guidelines

The chapter contains the following topics:

Reporting Server Instance

The Reporting Server instance(Informix dB) must be named as **cvp**, and must not be renamed.

Allow Only Reporting Users to Query Database

In the interests of security, allow only reporting users to generate reports.

Accurate Time Stamps for Reporting and Logging

Cisco Unified Customer Voice Portal (Unified CVP) components do not themselves synchronize machine times. However, customers must provide a cross-component time synchronization mechanism, such as Network Time Protocol (NTP), to ensure accurate time stamps for reporting and logging.

CPU-Intensive Reports

Do not run CPU-intensive reports off the database while the database is receiving data.

> **Note** Reports become more CPU intensive as the complexity associated with producing the report from the information available in the database increases. There is no sharp dividing line between intensive and non-intensive reports. The system performance must remain within the guidelines defined in the *Solution Design Guide for Cisco Unified Contact Center Enterprise*.

Database Backup and Recovery

Issues to keep in mind are:

• Managing your backup strategy

• Turning off the reporting server when doing database recovery

These issues are discussed in Database Backup, on page 110 and Database Recovery, on page 112.

Database Retention Settings

Ensure that the database is sized conservatively so that it never needs to emergency purge.

Data Security

Ensure data is secure by the following practices:

• Unified CVP offers administrators the ability to choose not to persist sensitive ECC data in the database. Users define ECC variables in Cisco Unified Intelligent Contact Manager Enterprise (Unified ICME) and by default they are not persisted in the Unified CVP database.

The Caller_input and FromExtVXML ECC variables are subject to many application-dependent uses. For security purposes, flag these two variables as not persistent in Unified ICME. If there is anything in them that must be stored, the routing script can copy the data to an applicable variable for storage in the database.

- Users can reduce the data generated by means of data filters (for VXML Server application detail data filtering. Either adding more exclusive filters, or using fewer inclusive filters, cuts down on the amount of data stored.

- Users can turn off logging of sensitive data containing caller's responses on a per-element basis. The caller's input, such as set of digits representing Social Security numbers or credit card numbers, can be set not to be logged, providing a security layer in case logs are compromised.

Database Sizing Issues

See the discussion in Data Categories and Data Retention, on page 107. Also see the *Solution Design Guide for Cisco Unified Contact Center Enterprise*.

Report Data Filtering Before Database Storage

Users can reduce the data generated by means of data filters. Either adding more exclusive filters, or using fewer inclusive filters, can cut down on the amount of data stored.

For information on filtering reporting data, see https://www.cisco.com/en/US/products/sw/custcosw/ps1006/products_user_guide_list.html.

Inclusive and Exclusive VXML Filters for Reporting

You use Inclusive and Exclusive VXML filters to control the data that the VXML Server feeds to the Reporting Server. Data feed control is crucial for:

Procedure

- Saving space in the reporting database.
- Preserving messaging communication bandwidth.

Inclusive and Exclusive Filter Configuration

To configure inclusive and exclusive filters for a Reporting Server:

Procedure

Step 1	Choose **Device Management** > **VXML Server**.
	The Find, Add, Delete, Edit VXML Servers window opens.
Step 2	You can search for a VXML Server by using the procedure in the Finding a VXML Server topic.
Step 3	From the list of matching records, choose the VXML Server that you want to edit.
Step 4	Click **Edit**.
	The VXML Server Configuration window opens to the General Tab.

Step 5 Select the **Configuration Tab**, then configure VXML Server properties.

Step 6 In the **VXML Applications Details: Filters** pane, enter an inclusive filter that defines the VXML elements to include in data sent to the Reporting Server.

Step 7 Optionally, enter an exclusive filter that excludes some of the data specified by the inclusive filter.

Step 8 When you finish configuring filters, click **Save** to save the settings in the Operations Console database or click **Save & Deploy** to save and apply the changes to the VXML Server.

Step 9 Shut down and then start the VXML Server and the primary and backup Call Servers.

Related Topics

VXML Inclusive and Exclusive Filter Rules

Inclusive and exclusive filters operate using the following rules:

- Filters are case sensitive.

- By default, all items but the Start, End, Subdialog_Start and Subdialog_End elements are filtered from reporting data unless they are added to an Inclusive Filter. The Subdialog_Start and Subdialog_End elements are never filtered from reporting data unless Reporting is disabled on the VXML Server.

- The Exclusive Filter takes precedence over the Inclusive Filter. For example, if an application name is in the Exclusive Filter, then all of the items of that applications are excluded from reporting data even if a particular field or element is listed in the Inclusive filter.

- The syntax for Inclusive/Exclusive filters is:

  `Appname.ElementType.ElementName.FieldName`

  or

  `AppName.*.*.SESSION:Varname`

  ✎

  Note This syntax is used to indicate session variables.

- A semicolon (;) should be used to separate each item in a filter. For example, `ElementA ; ElementB` is valid.
- A wildcard (*) can be specified anywhere within the application name, element type, element name, or field name.
- Element types, element names, and field names can contain alphanumeric characters, underscores, and a space character.
- An application name can contain alphanumeric characters and underscores, but the space character is not allowed. For example, `A_aa.B_bb.*C_cc_DD.E_ee_F*` is valid.

Element Types and Filter Names

The following table lists the various VXML element type and their flag.

Table 42: VXML Element Type and Filter Name

Element Type	Filter Name
VXML_TYPE_START	start
VXML_TYPE_END	end
VXML_TYPE_SUBDIALOG_START	SubdialogStart
VXML_TYPE_SUBDIALOG_RETURN	SubdialogReturn
VXML_TYPE_VOICE	voice
VXML_TYPE_DECISION	decision
VXML_TYPE_ACTION	action
VXML_TYPE_VXMLINSERT	vxml
VXML_TYPE_CUSTOM	custom
VXML_TYPE_HOTLINK	hotlink
VXML_TYPE_HOTEVENT	hotevent
VXML_TYPE_ELEMENTFLAG	flag_element
VXML_TYPE_REQICMLABEL	ReqICMLabel
VXML_TYPE_RECORDINGLABEL	RecordingLabel
VXML_TYPE_ELMENT_DEF	element_def

VXML Filter Wildcard Matching Example

The following table provides examples of VXML filter wildcard matching.

Table 43: Examples of VXML Filter Wildcard Matching

Filter	What It Matches
MyApplication.voice.*.*	Matches all voice elements in MyApplication
.voice..*	Matches all Voice elements in all applications.
MyApplication.*.*.var*	Matches all fields in MyApplication that start with with the string var.
MyApplication.*.*.*3	Matches all fields in MyApplication that end with 3.
MyApplication.*.*.SESSION:Company	Matches the Company session variable in MyApplication.

Inclusive and Exclusive VXML Filters for Reporting Example

The following table provides examples of some different combinations of Inclusive and Exclusive filters and the resulting data that the VXML Server feeds to the Reporting Server.

Table 44: Examples of Inclusive and Exclusive VXML Filters for Reporting

Inclusive Filter	Exclusive Filter	Data the VXML Server Feeds To the Reporting Server
`Application1.*.*.*`	None	All Application1 data

`Application1.*.*.*`	`*.*.Element1.*;`
	
	`*.*.Element2.*`
		All Application1 data, except Element1 and Element2

`Application1.*.*.*`	`*.*.Element1.*;`
	
	`*.*.Element2.*;`
	
	`*.*.*.Field1`
		All Application1 data, except Element1, Element2, and Field1

`Application1.*.*.*`	`*.voice.*.*` which matches Element3 and Element4	All Application1 data, except Element3 and Element4

`*.*.Element1.*;`
 	
`*.*.Element2.*;`
 	
`*.*.*.Field1`
	`Application1.*.*.*`	No data for Application1. Other data for other applications, such as Application2, which contain Element1, Element2 and Field1, will be fed.

`*.voice.*.*` which matches Element1, Element2, Element3, and Element4	`*.*.Element3.*;`
	
	`*.*.Element4.*`
		Only Element1 and Element2 and all applications.

`*.voice.*.*` which matches Element1 and Element2	`*.*.*.Field1`	Element1 and Element2, except for Field1, if it exists in those elements
`*.*.Element1.*`	None	Element1
`*.*.Element1.*`	`*.*.*.Field1`	Element1, except for Field1 if it exists in Element1

`*.*.*.Field1`	`*.*.Element3.*;`
	
	`*.*.Element4.*`
		Field1 in any elements except Element3 and Element4

A good strategy for using filters is to create an Inclusive filter that includes the data you want to save in the Reporting database and then create an Exclusive filter to *exclude* portions of the data, for example, sensitive security information such as Social Security Numbers. For example, you would:

- First, create an inclusive filter to include all information:

  `MyApp.voice.*.*`

- Then, create an exclusive filter to remove credit card and social security numbers information:

```
MyApp.voice.*.CreditCard; MyApp.voice.*.SSN
```

Informix, Operating System Time, and Local Time

Informix displays a datetime that corresponds to the same time zone as the Informix server operating system's time zone, represented in Universal Time Coordinated (UTC).

- If you wish a datetime to be displayed for a time zone other than that of the Informix server operating system, you must use reporting tools or SQL tools (for example, Java).

- If, in your system, you are using more than one Informix server for Unified CVP reporting, it is best if all such server operating systems are set to the same time zone. This helps avoid confusion.

Cisco Unified Customer Voice Portal and SQL Server Data Joining

To join data from a SQL server database and an Informix Database you must use a reporting tool that supports the ability to join data from two heterogeneous databases.

Reporting Password Policy Adherence

Reporting passwords are subject to both the Unified CVP password policy *and* the password policy enforced by the operating system of the computer on which the reporting server resides. For each aspect of the password, the Reporting password *must* meet the requirement of the more restrictive policy.

Related Topics

Passwords, on page 105

Purge and Backup Database Maintenance Tasks

The database backup and purge maintenance tasks are created as Windows Scheduled Tasks, and can be viewed in the Scheduled Tasks window (**Start** > **Programs** > **Accessories** > **System Tools** > **Scheduled Tasks**). These jobs log in as SYSTEM.

If the CVPDBNightlyPurge and CVPDBMidDayPurge tasks do not run, then the database will not be purged and will eventually become full, resulting in data loss.

If the CVPDBBackup task does not run the database will not be backed up.

Periodically, you should check the Scheduled Tasks to ensure the Last Run Time was as expected and there are no status messages.

Reporting Isolation Level

Reporting clients should never run with an isolation level of repeatable read because this could hold locks and prevent updates to the data.

Timestamp Synchronization

Call Servers, VXML Servers, and reporting servers must have their clocks synchronized to assure accurate timestamps in both the database and log files.

Since Unified CVP components do not themselves synchronize machine times, you must deploy a cross-component time synchronization mechanism, such as NTP.

Writing Efficient SQL Queries When Creating Reports

Keep these guidelines in mind:

- When writing SQL, developers must organize their WHERE clauses and put the most important join first. The most important join is the one that will reduce the size of the dataset to the least amount of rows.

- Write reports so that every field in the WHERE and ORDER BY clauses uses an indexed field.

- A subset of the data that satisfies any given query can protect the user and the database from generating massive data results. Including the word FIRST in Select statements will return only the amount of data requested. For example, `SELECT FIRST 1000 * FROM Call`.

- The second column in a composite index should never be used in a JOIN statement without the first column.

- Engineers writing database code should treat database, table, and column names as case sensitive--even though the current database is case insensitive--to ensure that the application is portable.

- Many operations hold database locks; therefore, reports should use a wait time of 30 seconds, if possible.

- It is possible to capture gigabytes of Unified CVP data in a single day. Any query against the database should target time ranges and subsets of data that will return in a reasonable time. Datetime columns are crucial selections. Sorting or grouping large quantities of data may exceed the capacity of the reporting server database as delivered.

- All sessions that connect to the reporting database should initiate with two statements:

  `SET ISOLATION DIRTY READ; SET LOCK MORE TO WAIT 30`.

  This prevents reporting queries from interfering with CallServer message persistence and improves the performance of reporting queries.

  ⚠

  Warning   Do not ever set the Isolation level to Repeatable Read.

- The internal ID generator limits the amount of total VXML subsystems to 8,000 per deployment.

> ⚠️ **Warning**   Do not ever write a SQL statement that selects into temp without specifying the 'no log' option.

Zero Duration Calls and Writing Reports

On occasion, messages are dropped. even for an otherwise successful call. In such cases, EndDateTime is set to the same value as StartDateTime. Thus, if a call appears to be of zero duration, report writers will know to exclude such a call from consideration in cases where it would otherwise skew metrics.