Interaction Concentrator 7.6

Deployment Guide

About Genesys

Genesys Telecommunications Laboratories, Inc., a subsidiary of Alcatel-Lucent, is 100% focused on software for call centers. Genesys recognizes that better interactions drive better business and build company reputations. Customer service solutions from Genesys deliver on this promise for Global 2000 enterprises, government organizations, and telecommunications service providers across 80 countries, directing more than 100 million customer interactions every day. Sophisticated routing and reporting across voice, e-mail, and Web channels ensure that customers are quickly connected to the best available resource—the first time. Genesys offers solutions for customer service, help desks, order desks, collections, outbound telesales and service, and workforce management. Visit www.genesyslab.com for more information.

Each product has its own documentation for online viewing at the Genesys Technical Support website or on the Documentation Library DVD, which is available from Genesys upon request. For more information, contact your sales representative.

Notice

Although reasonable effort is made to ensure that the information in this document is complete and accurate at the time of release, Genesys Telecommunications Laboratories, Inc., cannot assume responsibility for any existing errors. Changes and/or corrections to the information contained in this document may be incorporated in future versions.

Your Responsibility for Your System's Security

You are responsible for the security of your system. Product administration to prevent unauthorized use is your responsibility. Your system administrator should read all documents provided with this product to fully understand the features available that reduce your risk of incurring charges for unlicensed use of Genesys products.

Trademarks

Genesys, the Genesys logo, and T-Server are registered trademarks of Genesys Telecommunications Laboratories, Inc. All other trademarks and trade names referred to in this document are the property of other companies. The Crystal monospace font is used by permission of Software Renovation Corporation, www.SoftwareRenovation.com.

Technical Support from VARs

If you have purchased support from a value-added reseller (VAR), please contact the VAR for technical support.

Technical Support from Genesys

If you have purchased support directly from Genesys, please contact Genesys Technical Support at the following regional numbers:

Region	Telephone	E-Mail
North and Latin America	+888-369-5555 or +506-674-6767	support@genesyslab.com
Europe, Middle East, and Africa	+44-(0)-1276-45-7002	support@genesyslab.co.uk
Asia Pacific	+61-7-3368-6868	support@genesyslab.com.au
Japan	+81-3-6361-8950	support@genesyslab.co.jp

Prior to contacting technical support, please refer to the *Genesys Technical Support Guide* for complete contact information and procedures.

Ordering and Licensing Information

Complete information on ordering and licensing Genesys products can be found in the ***Genesys 7 Licensing Guide***.

Released by

Genesys Telecommunications Laboratories, Inc. www.genesyslab.com

Document Version: 76icon_dep_10-2008_v7.6.101.00

Table of Contents

List of Procedures

Preface

Welcome to the *Interaction Concentrator 7.6 Deployment Guide.* This document provides:

- A high-level overview of Interaction Concentrator 7.6, including architectural graphics.
- Deployment planning materials, including typical deployment scenarios.
- Deployment procedures, including troubleshooting tips.
- Metrics definitions, illustrated by basic call flows.
- An introduction to Interaction Database (IDB) schema, including sizing recommendations.

This document is valid only for the 7.6 release(s) of this product.

This preface provides an overview of this document, identifies the primary audience, introduces document conventions, and lists related reference information. It contains the following sections:

Interaction Concentrator collects and stores detailed data about the interactions and resources in customer interaction networks that use Genesys Framework (contact center, enterprise-wide, or multi-enterprise telephony and computer networks). Downstream reporting systems can access Interaction Concentrator data in near real time.

Intended Audience

This document, which is intended primarily for system integrators, assumes that you have a basic understanding of:

- Computer-telephony integration (CTI) concepts, processes, terminology, and applications.

- Network design and operation.
- Database design and operation.
- Your own network configurations.

You should also be familiar with:

- Genesys Framework architecture and functions.
- Genesys solutions deployed in your contact center.
- Your real-time and historical reporting objectives.

Chapter Summaries

In addition to this preface, this guide contains the following chapters:

- Chapter 1, "Product Overview," on page 15, describes the basic Interaction Concentrator architecture, the components of Interaction Concentrator and their functions, various deployment scenarios, and the functional roles that single or multiple Interaction Concentrator (ICON) instances play in a contact center. It also provides a high-level overview of Interaction Concentrator functionality, including features and functionality that are new in release 7.6.
- Chapter 2, "Deployment Planning," on page 31, lists the prerequisites for Interaction Concentrator deployment. It also provides other primary information that you need in order to plan an Interaction Concentrator installation, including information about compatibility with other Genesys components.
- Chapter 3, "Configuration and Installation," on page 39, describes the recommended deployment sequence for Interaction Concentrator. It also provides procedural steps for deploying each component in your Genesys environment.
- Chapter 4, "Starting and Stopping Interaction Concentrator," on page 57, describes the prerequisites for Interaction Concentrator startup and provides instructions for starting and stopping ICON.
- Chapter 5, "Configuration Options," on page 67, describes the configuration options that you can set up for your Interaction Concentrator through the ICON `Application` object and other configuration objects.
- Chapter 6, "Troubleshooting ICON Installation and Deployments," on page 117, provides solutions for the most common Interaction Concentrator problems encountered during startup.

Document Conventions

This document uses certain stylistic and typographical conventions—introduced here—that serve as shorthands for particular kinds of information.

Document Version Number

A version number appears at the bottom of the inside front cover of this document. Version numbers change as new information is added to this document. Here is a sample version number:

76icon_dep_05-2008_v7.6.001.00

You will need this number when you are talking with Genesys Technical Support about this product.

Type Styles

Italic

In this document, italic is used for emphasis, for documents' titles, for definitions of (or first references to) unfamiliar terms, and for mathematical variables.

Examples:

- Please consult the *Genesys 7 Migration Guide* for more information.
- *A customary and usual practice* is one that is widely accepted and used within a particular industry or profession.
- Do *not* use this value for this option.
- The formula, $x + 1 = 7$ where x stands for . . .

Monospace Font

A monospace font, which looks like `teletype or typewriter text`, is used for all programming identifiers and GUI elements.

This convention includes the *names* of directories, files, folders, configuration objects, paths, scripts, dialog boxes, options, fields, text and list boxes, operational modes, all buttons (including radio buttons), check boxes, commands, tabs, CTI events, and error messages; the values of options; logical arguments and command syntax; and code samples.

Examples:

- Select the `Show variables on screen` check box.
- Click the `Summation` button.
- In the `Properties` dialog box, enter the value for the host server in your environment.
- In the `Operand` text box, enter your formula.
- Click `OK` to exit the `Properties` dialog box.

- The following table presents the complete set of error messages T-Server® distributes in `EventError` events.
- If you select `true` for the `inbound-bsns-calls` option, all established inbound calls on a local agent are considered business calls.

Monospace is also used for any text that users must manually enter during a configuration or installation procedure, or on a command line:

Example:

- Enter `exit` on the command line.

Screen Captures Used in This Document

Screen captures from the product GUI (graphical user interface), as used in this document, may sometimes contain a minor spelling, capitalization, or grammatical error. The text accompanying and explaining the screen captures corrects such errors *except* when such a correction would prevent you from installing, configuring, or successfully using the product. For example, if the name of an option contains a usage error, the name would be presented exactly as it appears in the product GUI; the error would not be corrected in any accompanying text.

Square Brackets

Square brackets indicate that a particular parameter or value is optional within a logical argument, a command, or some programming syntax. That is, the parameter's or value's presence is not required to resolve the argument, command, or block of code. The user decides whether to include this optional information. Here is a sample:

```
smcp_server -host [/flags]
```

Angle Brackets

Angle brackets indicate a placeholder for a value that the user must specify. This might be a DN or port number specific to your enterprise. Here is a sample:

```
smcp_server -host <confighost>
```

Related Resources

Consult the following additional resources as necessary:

- The *Interaction Concentrator 7.6 User's Guide*, which will help you learn more about Interaction Concentrator functionality, including IDB architecture, models used in Interaction Concentrator, attached data processing, available stored procedures, and integration with other Genesys products.

- The *Interaction Concentrator 7.6 Physical Data Model* for your relational database management system (RDBMS) type, which will help you learn about IDB tables.
- The *Interaction Concentrator 7.6 Database Size Estimator*, which will help you estimate the size of your IDB when you are planning your deployment. The estimator is a Microsoft Excel spreadsheet available from the Genesys Technical Support website.
- The *Genesys 7 Hardware Sizing Guide*, which contains information about recommended hardware architectures and additional information related to database size estimation.
- *Genesys 7.6 Combined Log Events Help,* which describes the log events generated by every Genesys server application, including Interaction Concentrator.
- The documentation set for Genesys Info Mart release 7.6, if you intend to use Interaction Concentrator as a source of data for Genesys Info Mart.
- The documentation set for Genesys Outbound Contact release 7.6, if you intend to store outbound-related data in IDB.
- The documentation set for Genesys Universal Routing release 7.6, if you intend to store information about virtual queue usage in interaction processing in IDB.
- The documentation set for Genesys Multimedia release 7.6, if you intend to store interaction-related and related data about Multimedia interactions in IDB.
- The *Genesys Technical Publications Glossary,* which ships on the Genesys Documentation Library DVD and which provides a comprehensive list of the Genesys and CTI terminology and acronyms used in this document.
- The *Genesys 7 Migration Guide*, also on the Genesys Documentation Library DVD, which provides a documented migration strategy from Genesys product releases 6.x and higher to all Genesys 7.x releases. Contact Genesys Technical Support for additional information.
- The Release Notes and Product Advisories for this product, which are available on the Genesys Technical Support website at http://genesyslab.com/support.

Information about supported hardware and third-party software is available on the Genesys Technical Support website in the following documents:

- *Genesys 7 Supported Operating Systems and Databases*
- *Genesys 7 Supported Media Interfaces*

Genesys product documentation is available on the:

- Genesys Technical Support website at http://genesyslab.com/support.
- Genesys Documentation Library DVD, which you can order by e-mail from Genesys Order Management at orderman@genesyslab.com.

Making Comments on This Document

If you especially like or dislike anything about this document, please feel free to e-mail your comments to Techpubs.webadmin@genesyslab.com.

You can comment on what you regard as specific errors or omissions, and on the accuracy, organization, subject matter, or completeness of this document. Please limit your comments to the information in this document only and to the way in which the information is presented. Speak to Genesys Technical Support if you have suggestions about the product itself.

When you send us comments, you grant Genesys a nonexclusive right to use or distribute your comments in any way it believes appropriate, without incurring any obligation to you.

Document Change History

This section lists topics that are new in the current release of this document, or that have changed significantly from the preceding release.

New in Version 7.6.1

This deployment guide includes the following new features for the 7.6.1 release of Interaction Concentrator:

- Advanced memory management in order to support a large number of multimedia interactions. Three new configuration options help to optimize ICON's operational memory and support this feature:
  - `om-memory-optimization`, described on page 80
  - `om-max-in-memory`, described on page 80
  - `om-memory-clean`, described on page 114
- Filtering of extra Multimedia data in order to save IDB storage and improve ICON's overall performance. Two new configuration options enable this filtering feature:
  - `om-activity-report`, described on page 115
  - `om-check-filter-flag`, described on page 79
- A new purging mechanism for voice and Multimedia data that is stored in IDB. See “Stored Procedures” on page 18.
- Improved recognition of certain multi-site scenarios. A new configuration option regulates the mode that ICON uses for scenario recognition:
  - `gcti-mode-monitoring`, described on page 85

These features are described in more detail in the *Interaction Concentrator 7.6 User's Guide.*

Chapter

1 Product Overview

This chapter describes the basic Interaction Concentrator architecture, the components of Interaction Concentrator and their functions, various deployment scenarios, and the functional roles that single or multiple Interaction Concentrator (ICON) instances play in a contact center. It also provides a high-level overview of Interaction Concentrator functionality, including features and functionality that are new in release 7.6.

This chapter contains the following sections:

Basic Architecture

Interaction Concentrator is a Genesys product that collects and stores detailed data from various sources in a contact center that is empowered with Genesys software. Downstream reporting systems can access Interaction Concentrator data in near real time.

Operating on top of Genesys Framework, the Interaction Concentrator product consists of a server application called Interaction Concentrator (ICON) and a database called Interaction Database (IDB). The server receives data from the data sources such as Configuration Server, T-Server, or particular Genesys solutions; it then stores this data into IDB through Genesys DB Server.

Figure 1 depicts the basic ICON architecture, omitting the Framework components for the sake of simplicity.

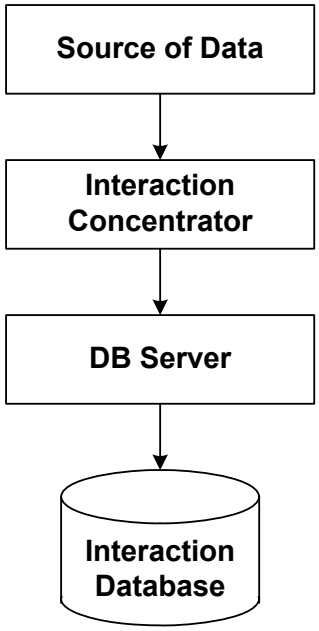

Figure 1: Basic Interaction Concentrator Architecture

Components and Functions

Interaction Concentrator consists of the following elements:

- ICON server
- Interaction Database

The following subsections describe each of these in turn.

ICON Server

The ICON server:

- Performs preprocessing of events received from Configuration Server, T-Server, Interaction Server, and Outbound Contact Server, according to the role configured for the ICON instance.

  Processing occurs in the in-memory queue (*accumulator*). You can configure the size of the in-memory queue or the interval at which data is written from it to the persistent queue.

  In ICON 7.6.1 you can also configure the total number of keep-in-memory interactions that can reside concurrently in an interaction queue or interaction workbin by setting memory configuration options on the ICON application. This functionality requires Interaction Server release 7.6.1 or higher. For more information about in-memory queue configuration options, refer to the option descriptions beginning on page 77.
- Prepares the data that will be stored in IDB.

- Writes the prepared data from the in-memory queue to the persistent queue. For more information about the persistent queue, see "Persistent Queue".
- Manages the data in the persistent queue.
- Writes data from the persistent queue into IDB.

For detailed information about the configuration options that determine ICON functionality and performance, see "Configuration Options" on page 67.

Persistent Queue

The persistent queue is a file that ICON creates and uses to store data before writing it to IDB. The persistent queue also stores information about database-writing requests to IDB. Data in the persistent queue survives a shutdown and restart of ICON.

Configuration Database Synchronization

In ICON 7.5, there is one persistent queue that stores data for all ICON instances and roles. In release 7.6, there is an additional persistent queue (`cfg-sync.db`) for the ICON instance that performs the `cfg` role. This queue plays an important role in maintaining IDB synchronization with the Configuration Database. ICON keeps a timestamp in the persistent queue for configuration data changes and, on startup, requests from Configuration Server all configuration changes that occurred after that timestamp. For recommendations about best practices regarding synchronization, see the chapter about resynchronization in the *Interaction Concentrator 7.6 User's Guide*.

Persistent Queue Configuration Options

ICON configuration options enable you to specify:

- The file name of the persistent queue for all roles except the `cfg` role. The name of the persistent queue for the ICON instance with the `cfg` role is `cfg-sync.db` and it cannot be changed.
- The frequency (in terms of number of committed transactions) with which ICON clears data out of the persistent queue.
- Thresholds for environment failure alarms.
- Persistent queue behavior at startup.

The alarm thresholds can also be used to monitor ICON performance. For more information about the persistent queue configuration options, see the options starting on page 82.

Note: The size of the persistent queue is not formally limited by ICON, but the operating system may impose some limitations.

ICON Server Interfaces

The ICON server interfaces with:

- Solution Control (Local Control Agent [LCA]), to control when the ICON server starts and stops.
- Configuration Server, to read Interaction Concentrator application configuration options and other configuration objects and options that affect Interaction Concentrator functionality. (This interface is logically separate from ICON's connection to Configuration Server as a source of data about contact center resources—see "Sources of Data" on page 19.)
- Message Server, to log messages to the Central Logger.

Note: Interaction Concentrator does not support the use of the Transport Layer Security (TLS) protocol to secure data exchange between the components with which the ICON server interfaces.

Interaction Database

The Interaction Database stores data about contact center interactions and resources at a granular level of detail. IDB is a database optimized for storage (in other words, for inserting data). Interaction Concentrator itself does not provide a reporting facility. You can use IDB as a consistent and reliable data source for downstream reporting applications.

For a high-level description of the IDB architecture, see the chapter about IDB schema in the *Interaction Concentrator 7.6 User's Guide.* For a complete table structure and descriptions of all IDB tables and fields, see the *Interaction Concentrator 7.6 Physical Data Model* document for your particular RDBMS.

Stored Procedures

Interaction Concentrator uses a number of stored procedures. Most of these are totally internal to Interaction Concentrator functioning, and therefore they are not relevant to end users. However, the following stored procedures do require user input or action:

Merge

- `gsysIRMerge` and `gsysIRMerge2`—The merge procedure that finalizes data processing of closed single-site and multi-site interactions.

Purge

- Use either of the two sets of purge procedures:
  - `gsysPurgeIR`, `gsysPurgeUDH`, `gsysPurgeLS`, and `gsysPurgeOS` to purge voice interactions, user data history, agent login session, and outbound data safely from IDB.
  - `gsysPurge76` and `gsysPurgeOS` to purge voice, multimedia, open media, attached data, agent login session, and outbound interaction data safely from IDB.

Time-Setting

- `gsysInitTimeCode`—The stored procedure that populates the `G_TIMECODE` table, to enable time-interval reporting.

Custom Dispatchers

- `gudCustDisp1` and `gudCustDisp2`—The stored procedures that are used to customize attached data processing.

For more information about these stored procedures, refer to the chapter about special stored procedures in the *Interaction Concentrator 7.6 User's Guide.*

Sources of Data

Table 1 summarizes the sources from which ICON collects data. For Interaction Concentrator 7.5 and 7.6, the range of types of data and sources is wider than for Interaction Concentrator 7.2.

Table 1: ICON Sources of Data

Type of Data	Source	Applicable Interaction Concentrator Release		
		7.2	7.5	7.6
Configuration data for your contact center resources	Configuration Server	Yes	Yes	Yes
Detailed CTI–related data about call activity in your contact center	T-Server	Yes	Yes	Yes
Detailed CTI-related data about Voice over IP (VOIP) interaction activity in your contact center	T-Server (SIP Server)	No	No	Yes
Detailed data about Open Media interaction activity (including e-mail and non-SIP chat) in your contact center	Interaction Server	No	Yes	Yes
Detailed data about SIP chat interaction activity in your contact center	SIP Server	No	Yes	Yes
Media types for Open (custom-defined) Media	Interaction Server	No	No	Yes
Detailed data about virtual queue usage in interaction processing	• For voice calls: T-Server (from Universal Routing Server [URS]) • For Multimedia interactions: Interaction Server (from URS)	No	Yes	Yes
Data specific to outbound calls and campaigns	OCS	Yes	Yes	Yes

Deployment Scenarios

The Interaction Concentrator architecture is flexible enough to store reporting data for a contact center environment of practically any size. This section provides basic deployment scenarios and describes how many components you would use in each.

Genesys provides an interactive tool to help you estimate the required size of IDB, based on details you provide about projected agent activity, outbound activity, ICON server and database settings, and user data. This tool, the *ICON 7.6 Database Size Estimator,* is a Microsoft Excel spreadsheet that is available from the Genesys Technical Support website. See also the Interaction Concentrator chapter in the *Genesys 7 Hardware Sizing Guide*.

The architectural choice for your contact center depends on your resources and reporting requirements. In fact, you can tailor the basic scenarios described in this section to fit the needs of your contact center at the lowest cost. For example, you can deploy a single instance of ICON for a subset of T-Servers (as opposed to a one-to-one correspondence in the number of ICON and T-Server instances). Alternatively, you can keep data for a certain site in a separate IDB, if it is not necessary to include data from this site in a consolidated report.

For additional considerations that may affect your choice of deployment, see "Recommended Role Assignment" on page 26.

Scenarios

This section provides general descriptions of the following scenarios:

- Single-site deployment (see page 21)
- Multi-site deployment (see page 21)
- Network deployment (see page 24)

Diagram Conventions

To simplify the deployment diagrams in this section:

- DB Server, which enables a connection between ICON and IDB, is omitted from the diagrams, even though it is required in actual deployments (see Figure 1 on page 16).
- Storage of configuration data is not shown, even though it is required in actual deployments.
- Storage of outbound-related call data is not shown because it is optional. It is described separately in the chapter about integrating with Outbound Contact in the *Interaction Concentrator 7.6 User's Guide.*
- Interaction Server is not shown for solutions that include Multimedia. Notes in the text indicate the deployment scenarios that are suitable for Multimedia solutions (see "One IDB" and "One ICON and One IDB per Site" on page 22). In these solutions, the Interaction Server occupies the same position in the architecture as a T-Server.

Single-Site Deployment

In a single-site contact center, two approaches to Interaction Concentrator deployment are most typical:

- A single Interaction Database
- Redundant instances of Interaction Database

One IDB

The simplest deployment scenario, which is suitable for smaller, single-site contact centers, consists of a single ICON instance that stores all data into a single IDB instance, as shown in Figure 2.

Note: Figure 2 illustrates the deployment for voice interactions. This type of deployment is also suitable for Multimedia solutions—the Interaction Server occupies the same position in the architecture as T-Server.

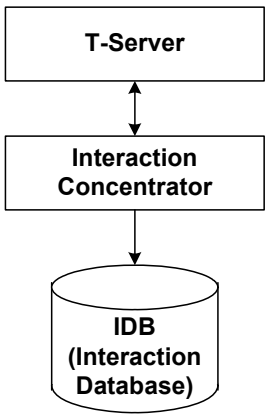

Figure 2: Single-Site Deployment: A Single IDB Instance

Multi-Site Deployment

In a multi-site contact center, approaches to Interaction Concentrator deployment vary, depending on network delays between sites, the need for across-the-sites reporting, and other considerations. The following is the basic list of deployments to consider:

- A single ICON instance and a single IDB instance per site (see page 22)
- A single ICON instance and a single, centralized IDB for the entire contact center (see page 22)
- Multiple ICON instances and a single, centralized IDB for the entire contact center (see page 23)

The following subsections describe each of these deployments in turn.

See also “Network Deployment” on page 24.

One ICON and One IDB per Site

In a multi-site deployment with a single instance of ICON and a single instance of IDB in each site, each IDB is populated independently from the other with CTI-related data from the T-Server that serves that site (see Figure 3).

Note: Figure 3 illustrates the deployment for voice interactions. This type of deployment is also suitable for Multimedia solutions—the Interaction Server occupies the same position in the architecture as T-Server. Genesys recommends that you include only one Interaction Server in your deployment.

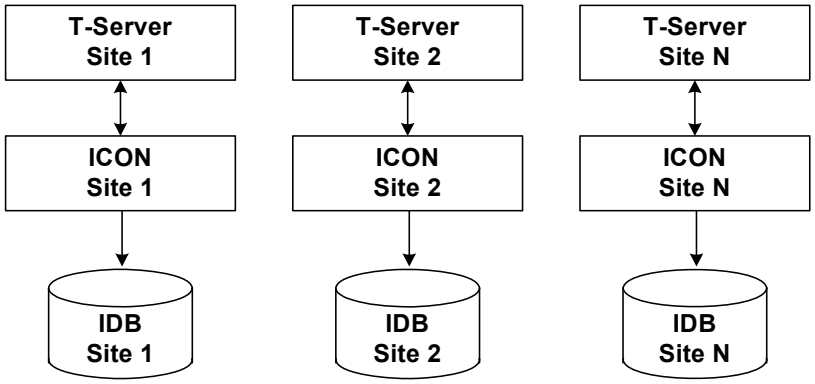

Figure 3: Multi-Site Deployment: Independent IDB Instances

Although the data for a particular site is readily available, this deployment does not provide across-the-sites reporting data for the entire contact center. Merging of data between IDBs is the responsibility of the downstream reporting application. Reporting of multi-site calls is also limited by the visibility of those calls at a particular site.

One ICON and One IDB per Contact Center

In a multi-site deployment with a single ICON instance and a single, centralized IDB instance, call details from all contact center sites come into the same database through the same ICON (see Figure 4).

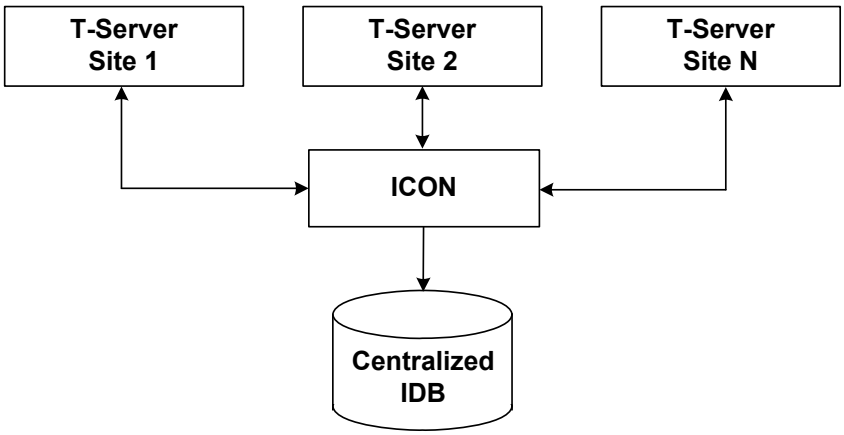

Figure 4: Multi-Site Deployment: A Single ICON and a Centralized IDB Instance

This scenario helps you avoid the need to merge data from different databases. However, be aware of the following:

- You must regularly run the Interaction Concentrator intra-IDB merge stored procedure to ensure correct reporting of multi-site calls (see "gsysIRMerge and gsysIRMerge2" in the *Interaction Concentrator 7.6 User's Guide*).
- Network delays might impact the timeliness of data availability.
- ICON performance is negatively affected during high-peak hours, when each T-Server handles high call volume.

Multiple ICONs and One IDB per Contact Center

In a multi-site deployment with a separate ICON instance at each site and a single, centralized IDB instance, call details from all contact center sites come into the same database through separate ICONs (see Figure 5).

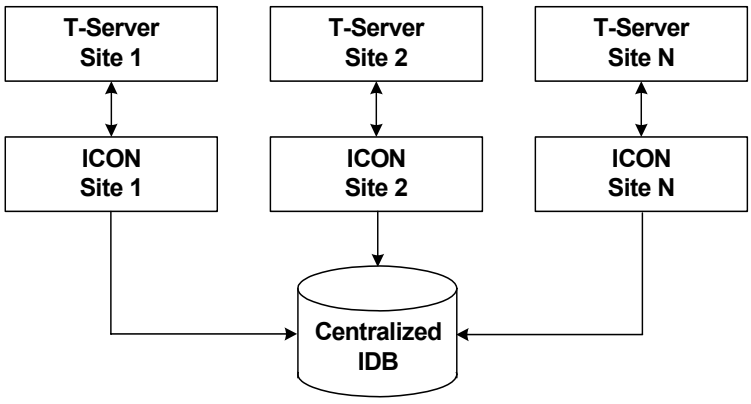

Figure 5: Multi-Site Deployment: Multiple ICONs and a Centralized IDB Instance

Like the scenario of one ICON and one IDB for the contact center (see Figure 4), this deployment provides the benefit of recording all contact center data in the same database.

However, this scenario provides the additional benefit of improved ICON performance, because a single ICON instance does not require a connection to every T-Server in the contact center. In addition, because T-Server and ICON instances are co-located at a particular site, network delays between these components are minimal.

Nevertheless, the effectiveness of data storage to IDB still depends on network delays between a given ICON instance and IDB, as well as on the performance of your RDBMS. Also, to ensure data correctness for multi-site calls, you must regularly run the Interaction Concentrator intra-IDB merge stored procedure (see "gsysIRMerge and gsysIRMerge2" in the *Interaction Concentrator 7.6 User's Guide*).

Network Deployment

In a network configuration, a number of T-Server applications are connected to a network T-Server. The ICON instance connects to the network T-Server (see Figure 6).

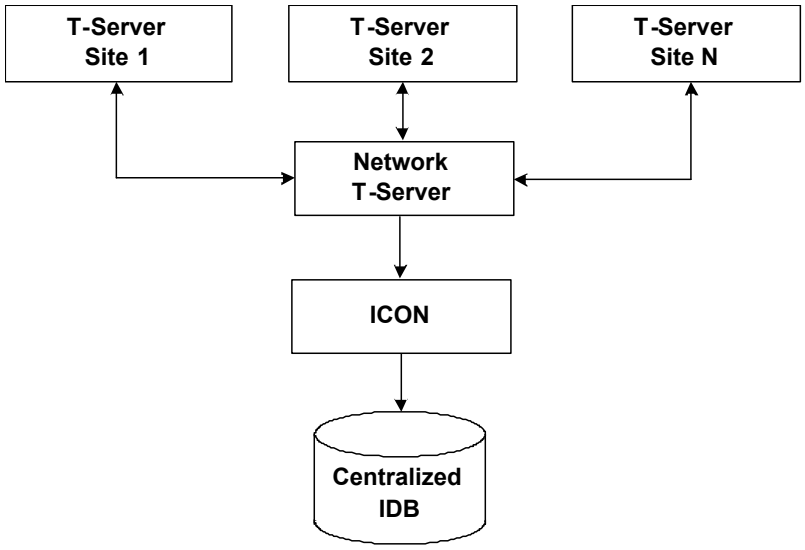

Figure 6: Network Deployment: A Single Network T-Server, a Single ICON, and a Centralized IDB Instance

From the Interaction Concentrator point of view, a network deployment is essentially a variant of either the single-site deployment or, where multiple network T-Servers are used (for example, for load-balancing), the multi-site deployment in which there are multiple ICONs and one IDB per contact center. The important limitation is that only one network T-Server can be connected to an ICON instance.

ICON Roles

In a contact center that has a large Genesys configuration environment, and/or that processes high call volumes, possibly with large amounts of attached data, you can improve Interaction Concentrator performance by deploying multiple ICON instances, each of which collects data only of a certain type.

The following are the possible types of data that you can request a given ICON instance to store, in any combination:

- Service information—All ICON instances are required to store service information in IDB for identification purposes, regardless of configuration settings.
- Configuration information—An ICON instance stores the initial contact center configuration state and a history of configuration changes that it retrieves from Configuration Server. Depending on the deployment scenario, the ICON instance can store configuration information about the contact center as a whole or, in a multi-tenant configuration environment, about individual tenants.
- Interaction-related and party-related information—An ICON instance can store T-Server data that pertains to calls and the parties (connections) associated with those calls. In a Multimedia solution, ICON stores similar Interaction Server data about Multimedia interactions (e-mail and chat).

> **Note:** If T-Server is present on the `Connections` tab of the ICON `Application` object, ICON will perform the `gcc` role, regardless of whether you have configured the ICON instance to perform this role. However, for an ICON instance to be able to process data from Interaction Server, it must have the `gcc` role defined (see “role” on page 76 for more information).

- Agent state and login session state information—An ICON instance can store T-Server and, if applicable, Interaction Server data that pertains to agent states and agent login sessions.
- Attached data information—An ICON instance can store T-Server and, if applicable, Interaction Server data that pertains to the attached data that is associated with interactions.
- Outbound calls information—In an environment with the Genesys Outbound Contact solution, an ICON instance can store OCS data that pertains to outbound calls and campaigns.

In the example shown in Figure 7, the ICON instance named `ICON 1` handles only the history of configuration changes (configuration data) from Configuration Server. The instance named `ICON 2` handles the business data that agents attach to calls and that T-Server includes in TEvents that pertain to those calls (attached data), as well as any other CTI-related data from the same T-Server. Finally, the instance named `ICON 3` handles OCS data only.

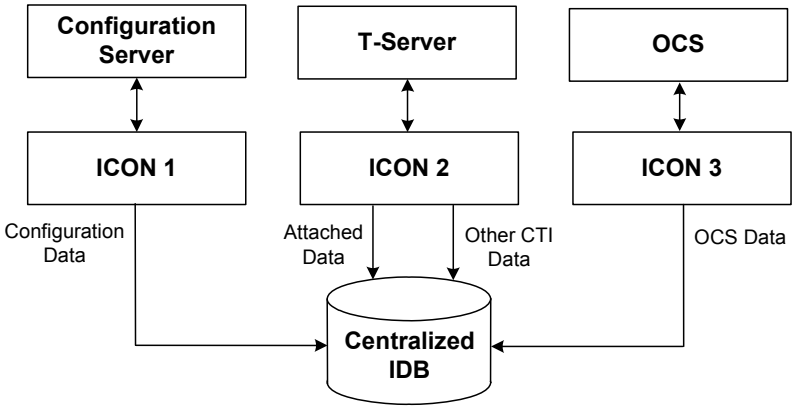

Figure 7: Role Division Among ICON Instances

Recommended Role Assignment

Genesys recommends the following role assignments in an environment with multiple ICON instances:

- Do not distribute call-related and party-related information, agent state and login-session state information, and attached data information among separate ICON instances. Assign a single ICON instance to write these three data types from a single T-Server.
- When multiple ICON instances are writing data to a particular IDB, assign only one of these instances to write configuration data to this IDB.
- In large configuration environments, Genesys recommends dedicating one of the ICON instances to process configuration data (`role = cfg`) and disabling configuration data processing in the other ICON instances (`role = ~cfg`). This improves ICON performance on startup, because the initial configuration loading stage can take quite a long time.

For the values that enable role assignments, refer to the description of the `role` configuration option on page 76.

Supported Features and Functionality

In addition to new functionality described in "New in This Release" on page 28, and depending on the role configured for the Interaction Concentrator instance, Interaction Concentrator provides the following features and functionality to support reporting about contact center activities:

- Captures and stores information about the current contact center configuration (objects and associations), and preserves information about deleted configuration objects and terminated associations.

- Captures and stores detailed information about active and completed voice interactions, including switch, DN, time, and routing information about calls and parties. Interaction Concentrator uses a globally unique call identifier.
- Captures and stores detailed information about Multimedia interactions (e-mail and chat). For more information, see the chapter about integrating with Genesys Multimedia in the *Interaction Concentrator 7.6 User's Guide.*
- Captures and stores detailed information about agent states and login sessions, for agents handling voice as well as Multimedia (e-mail and chat) interactions. For more information, see the chapter about agent states and login sessions in the *Interaction Concentrator 7.6 User's Guide.*
- Supports custom agent states, for agents handling voice interactions. For more information, see the chapter about custom states and common data in the *Interaction Concentrator 7.6 User's Guide.*
- For all types of interactions, captures and stores detailed information about virtual queue usage in interaction processing. For more information, see the chapter about monitoring virtual queues in the *Interaction Concentrator 7.6 User's Guide.*
- For voice interactions, captures and stores detailed information about virtual routing point (VRP) usage in call processing.
- Captures and stores detailed information about interactions that are generated in a network-based contact solution environment.
- Captures and stores detailed information about interactions that are generated in a network call parking environment.
- Stores attached data and captures the history of attached data changes for voice interactions as well as Multimedia (e-mail and chat) interactions. For more information, see the chapter about attached data in the *Interaction Concentrator 7.6 User's Guide.*
- Supports customized attached data processing for voice calls. For more information, see the chapter about attached data in the *Interaction Concentrator 7.6 User's Guide.*
- Captures and stores detailed information about outbound campaigns, including:
  - History of campaign processing
  - History of chain processing
  - Precalculated metrics provided by OCS

  For more information, see the chapter about integrating with Outbound Contact in the *Interaction Concentrator 7.6 User's Guide.*
- Supports real-time, intraday reporting by writing data to IDB as soon as the data is available (as opposed to after the interaction is completed).

- Provides a sophisticated recognition mechanism, utilizing Inter-Site Call Linkage (IS-Links), to process multi-site interactions. ICON receives information from T-Server regarding the relationship between a given interaction and an interaction at a different site. As a result, complete data is available for reporting across sites. Interaction Concentrator provides a stored procedure to merge the interaction records for multi-site interactions. For more information about the merge procedure, see the chapter about special stored procedures in the *Interaction Concentrator 7.6 User's Guide.*
- Supports multibyte character encoding.
- Stores time information in two formats:
  - Greenwich Mean Time (GMT)—As a `datetime` data type.
  - Coordinated Universal Time (UTC) seconds—As an `integer` data type.

  ICON obtains the time information from the timestamps of the data provider events (for example, T-Server TEvents), in the form of UTC seconds.

New in This Release

The 7.6.1 release of Interaction Concentrator includes the following new features:

- An enhancement to support a large number of concurrently active Multimedia interactions.
- A purging mechanism for Multimedia data that is stored in IDB.
- Improved performance for purging voice data that is stored in IDB using the new purging mechanism.
- The ability to filter out some Multimedia data that is not relevant for reporting, in order to save IDB storage space and improve overall performance.

The 7.6 release of Interaction Concentrator provides the following additional or changed functionality:

- Provides the ability to resynchronize the configuration data in Interaction Database (IDB) with Configuration Database on demand. For more information, refer to the *Interaction Concentrator 7.6 User's Guide.*
- Provides a configurable filtering mechanism for certain types of data to enable the optimization of database size and performance.
- Supports high availability of data at the extraction, transformation and loading (ETL) level by providing data redundancy through the use of two or more IDBs. In addition to the previously available HA of voice details and outbound data, the applicable data now includes configuration data, agent-related data, and SIP chat data. For more information, refer to the *Interaction Concentrator 7.6 User's Guide.*

- Supports media types for Open Media in all the areas where e-mail and chat data (other than related to SIP) were previously supported.
- Supports reporting on SIP chat.
- Provides an ability to identify if a chat session has the focus.
- Provides an ability to report after-call work (ACW) for the first interaction associated with ACW, for voice interactions.
- Provides the ability to suppress, for voice interactions, the interruption of the `ACW` and `NotReady` agent states by interactions coming to, or produced by, the agent.
- Supports a wider range of detailed routing results for interactions that are distributed from virtual queues.

Chapter

2 Deployment Planning

This chapter lists the prerequisites for Interaction Concentrator deployment. It also provides other primary information that you need in order to plan an Interaction Concentrator (ICON) installation, including information about compatibility with other Genesys components.

This chapter contains the following sections:

Compatibility

This section lists the various Genesys components with which Interaction Concentrator release 7.6 is compatible.

For information about supported operating systems and relational database management systems (RDBMSs), see the *Genesys 7 Supported Operating Systems and Databases* document. Note that Oracle versions 8.1 and earlier are not supported.

Table 2 lists the Genesys product components with which Interaction Concentrator operates. Refer to the *Interaction Concentrator 7.6 Release Notes* for any updates to the release requirements for the various components.

Table 2: Interaction Concentrator Compatibility

Area of Functionality	Component/Product
Configuration Layer	• Configuration Server release 7.2 or higher • DB Server release 7.2 or higher
Management Layer	• Message Server release 7.1 or higher • Local Control Agent release 7.1 or higher
T-Server	• T-Server release 7.2 or higher
Multimedia	• Multimedia Interaction Server release 7.5 or higher
Outbound Contact	• Outbound Contact release 7.2 or higher **Note:** If you use OCS 7.2 as the data source, attached data is not automatically linked to the call record. You must specially configure attached data to be linked.
Routing	• Universal Routing Server release 7.2 or higher **Note:** For virtual queue reporting, the minimum required version of Universal Routing Server is release 7.2.001.11.

Prerequisites

Interaction Concentrator has important specific requirements in the following areas:

- Hosting (see page 33)
- Genesys Framework (see page 33)
- Outbound Contact Solution (see page 34)
- Universal Routing Solution (see page 35)
- Multimedia (see page 35)
- Interaction Database (see page 36)

The following subsections describe each of these areas in turn. Before you install Interaction Concentrator, review the requirements and recommendations in these subsections.

Hosting

Genesys recommends that you or your IT specialist assign host computers to Genesys software before you start a Genesys installation. Keep in mind the following restrictions:

- Do not install all Genesys server applications on the same host computer.
- When installing multiple server applications on the same host computer, prevent all of them, except Configuration Server, from using swap space.

See the "Network Locations for Framework Components" chapter of the *Framework Deployment Guide* and the "Architecture" chapter of the *Framework Management Layer User's Guide,* for information about the optimal locations for:

- Configuration Layer components
- Management Layer components
- T-Server

For Interaction Concentrator and its DB Server, observe the following recommendations:

- Install DB Server on the same computer as the Interaction Database (IDB).
- You can use the same host computer for ICON and T-Server.

Time Synchronization Among Hosts

In an environment with either a single ICON instance or multiple ICON instances operating with multiple T-Servers, synchronize the system time on the T-Server host computers to one second or better.

Genesys Framework

Deploy the Genesys Framework components before you deploy Interaction Concentrator.

Configuration Layer

At the very least, you must set up the Configuration Layer of Genesys Framework. You cannot configure Interaction Concentrator components without the Configuration Layer. This layer contains DB Server, Configuration Server, Configuration Database, Configuration Manager, and, optionally, Deployment Wizards.

You will need a Genesys Management Framework product CD in order to install components of the Configuration Layer.

For information about, and deployment instructions for, these Framework components, see the *Framework Deployment Guide.*

Management Layer

If you intend to monitor or control Interaction Concentrator and its DB Server through the Management Layer, you must also configure and install

Management Layer components—in particular, LCA, Message Server, Solution Control Server (SCS), and SCI.

To monitor the status of Interaction Concentrator components through the Management Layer, you must load an LCA instance on every host that is running ICON and DB Server instances. Without LCA, the Management Layer cannot monitor the status of these components. If you do not use the Management Layer, you do not need LCA.

You will need a Genesys Management Framework product CD in order to install the components of the Management Layer.

For information about, and deployment instructions for, these Framework components, see the *Framework Deployment Guide* and the *Framework Management Layer User's Guide.*

Telephony Objects

Create configuration objects for every PBX about which you want Interaction Concentrator to store data.

Use Configuration Manager to configure telephony objects, including a `Switching Office` object and `Switch` object for the PBX, and one `DN` (Directory Number) object for each user's telephone number. For configuration settings that are specific to Interaction Concentrator, see "Switch Configuration Options" on page 103 and "DN Configuration Options" on page 112.

T-Server

If you intend to collect computer-telephony integration (CTI)–related (call) reporting data, configure and install a T-Server application for your particular PBX, if it is not yet deployed. Make sure that the `Switch` object that this T-Server will serve is specified in the `T-Server Application Properties` dialog box.

In a multi-site environment, deploy one T-Server application for each PBX.

Note: All T-Servers from which an ICON instance should collect data must be listed among the ICON `Application` object's connections.

You will need a Genesys Media product CD in order to install your T-Server.

For information about, and deployment instructions for, telephony objects and T-Server, see the *Framework T-Server Deployment Guide* for your particular T-Server.

Outbound Contact Solution

If you intend to collect data about outbound campaigns, deploy components of the Genesys Outbound Contact solution, release 7.2 or higher. If you have an earlier release of Outbound Contact, upgrade to release 7.2 before you deploy Interaction Concentrator.

To provide outbound information to ICON, at least one OCS application must exist and be properly configured.

Note: All OCS instances from which an ICON instance should collect data must be listed among the ICON `Application` object's connections.

For deployment instructions for Outbound Contact components, see the *Outbound Contact Deployment Guide.* For Outbound Contact migration instructions, see the latest *Genesys 7 Migration Guide.* For recommendations on how to enable outbound reporting in Interaction Concentrator, refer to the chapter about integrating with Outbound Contact in the *Interaction Concentrator 7.6 User's Guide.*

Universal Routing Solution

If you intend to collect data about virtual queues, deploy components of the Universal Routing solution, release 7.2 or higher, that support virtual queue functionality. If you have an earlier release of Universal Routing, upgrade to a release that supports virtual queue functionality.

Note: Interaction Concentrator requires Universal Routing Server (URS) release 7.2.001.11 higher for virtual queue reporting.

In order to provide virtual queue information to Interaction Concentrator, at least one URS application must exist.

New functionality in Interaction Concentrator 7.6 related to storing, in IDB, extended routing results from virtual queues, requires Universal Routing Server (URS) release 7.6. You must also set `report_reasons` and `report_targets` configuration options in URS to extract this extended routing information.

For deployment instructions for Universal Routing components, see the *Universal Routing Deployment Guide.* For Universal Routing migration instructions, see the latest *Genesys 7 Migration Guide.* For recommendations on how to enable virtual queue reporting in Interaction Concentrator, refer to the chapter about monitoring virtual queues in the *Interaction Concentrator 7.6 User's Guide.*

Multimedia

If you intend to collect interaction, agent state, and agent login session data for Multimedia interactions (e-mail and chat), configure and install an Interaction Server application for your Multimedia switch, if it is not yet deployed. Make sure that the `Switch` object that this Interaction Server will serve is specified in the `Interaction Server Application Properties` dialog box.

In a multi-site environment, deploy one Interaction Server application for each Multimedia switch.

Note: The Interaction Server from which an ICON instance should collect data must be listed among the ICON `Application` object's connections.

ICON release 7.6.1 introduces new functionality to support a large number of concurrently active Multimedia interactions. This functionality requires Interaction Server release 7.6.1 or higher.

For information about—and deployment instructions for—Interaction Server, see the *Multimedia Deployment Guide*. For recommendations on how to enable Multimedia reporting in Interaction Concentrator, refer to the chapter about integrating with Genesys Multimedia in the *Interaction Concentrator 7.6 User's Guide.*

Interaction Database

Interaction Concentrator uses IDB to store reporting data. At least one IDB instance is required, which can be running on any Genesys-supported RDBMS except Sybase and Informix (release 7.6 only). Oracle versions 8.1 and earlier are also not supported. (For a list of Genesys-supported RDBMSs, see *Genesys 7 Supported Operating Systems and Databases*.)

When planning an installation, observe the following recommendations for IDB:

- Review the information about the IDB structure in the chapter about IDB schema in the *Interaction Concentrator 7.6 User's Guide.*
- Estimate IDB size, using the *ICON 7.6 Database Size Estimator.* (For more information, see "Sizing the Interaction Database" on page 37.)

General Considerations

As described in Chapter 1 on page 15, Interaction Concentrator is flexible enough to fit any contact center. When planning a deployment, evaluate your environment and your reporting needs. Review the main deployment scenarios in "Deployment Scenarios" on page 20, and answer the following questions:

- How many ICON applications would you use, and what data would each ICON instance handle? From what sources would the data come to a given ICON instance?
- How many Interaction Databases would you use, and what data would each IDB instance store? Which ICON instances would store the data into a particular IDB instance? In the case of multiple IDB instances, would you need to deploy a centralized IDB, and, if so, from which subset of IDBs would data be merged into the centralized IDB? How often would the merge procedure be run?

- How many DB Server applications would you use, if you deploy multiple IDB instances? Would any of these DB Server instances handle database requests for servers other than ICON?
- How many Database Access Point (DAP) applications would you use? What data would a particular ICON instance store through a particular DAP?

The answers to these questions will help you determine the Interaction Concentrator deployment topology and the main configuration settings for all components.

Sizing the Interaction Database

The size of your IDB depends on your deployment scenario, including such factors as typical call flows, attached data storage, values configured for storing outbound data in custom or secure fields, and the amount of time that records will be retained in the database.

Genesys provides an interactive tool to help you estimate the required size of your IDB. This tool, the *Interaction Concentrator 7.6 Database Size Estimator*, is a Microsoft Excel spreadsheet that is available on the Genesys Technical Support website.

For more information about database sizing and deployment guidelines, see the Interaction Concentrator section in the *Genesys 7 Hardware Sizing Guide*.

Chapter

3 Configuration and Installation

This chapter describes the recommended deployment sequence for Interaction Concentrator. It also describes how to deploy each component in your Genesys environment.

This chapter contains the following sections:

Note: The information in this chapter is primarily directed towards first-time deployments of Interaction Concentrator. If you are migrating from an earlier release of Interaction Concentrator, ensure that you review the Interaction Concentrator chapters in the *Genesys 7 Migration Guide* before you perform any installation procedures.

Recommended Deployment Order

Before you deploy Interaction Concentrator, review the information in Chapter 2 on page 31, and ensure that you have accounted for all prerequisites for the installation.

Then, deploy Interaction Concentrator in the following order:

1. `Host` configuration objects

   Use Configuration Manager to configure a `Host` configuration object for the computers on which the DB Server and Interaction Concentrator (ICON) server applications will reside.

For information about Genesys configuration objects, see the *Framework 7.6 Configuration Manager Help.*

2. Telephony objects

   Use Configuration Manager to make any modifications to the telephony objects on which ICON will report, including the `Switch` object for the PBX and any `DN` (Directory Number) objects that are configured for this `Switch` object. For configuration settings that are specific to ICON, see "Switch Configuration Options" on page 103 and "DN Configuration Options" on page 112.

3. DB Server

   If you decide to use a DB Server that serves another solution for Interaction Concentrator storage purposes, skip this step.

   To configure and install a DB Server that will handle ICON requests for IDB data storage, use standard deployment instructions from the *Framework 7.6 DB Server User's Guide.* (See also Deploying DB Server for the major steps in the procedure.)

   For performance reasons, Genesys recommends that you set up the DB Server on the same box as the RDBMS server.

   In an environment with multiple IDB instances at separate sites, deploy one DB Server per IDB. In an environment with multiple IDB instances at the same site, deploying one DB Server for all IDB instances is sufficient.

   You will need a Management Framework 7.6 product CD in order to install DB Server.

4. Interaction Concentrator

   Configure and install Interaction Concentrator, as described in Deploying Interaction Concentrator, page 42.

   In an environment with multiple ICON instances, repeat the steps in Deploying Interaction Concentrator, page 42 for each ICON instance, making the necessary adjustments when you configure ICON connections and configuration options.

   You will need an Interaction Concentrator 7.6 product CD in order to install Interaction Concentrator.

5. Interaction Database

   Install a database for ICON data on one of the supported RDBMSs and initialize IDB, as described in "Creating IDB" on page 49.

   In an environment with multiple IDB instances, repeat the steps in "Creating IDB" on page 49 for each IDB instance.

6. Database Access Point

   Configure a `DAP Application` object that specifies IDB connection parameters, as described in Configuring a DAP, page 54.

In an environment with multiple ICON instances, each of which stores data to its own IDB instance, repeat the steps in Configuring a DAP, page 54 to create a separate DAP for each IDB instance. If you decide to write different types of data from a single ICON instance to different databases, also configure a separate DAP for each database.

Ensure that the `role` options that you specify for the DAP are consistent with the `role` options specified for the ICON instance that it serves.

Deploying DB Server

After you configure `Host` objects for Interaction Concentrator components, deploy as many instances of DB Server as you need using the following procedure.

Procedure: Deploying DB Server

Purpose: To configure and install as many instances of DB Server as you need.

Prerequisites

- Configure `Host` objects for Interaction Concentrator components. See “Hosting” on page 33.

Start of procedure

1. Import the application template for DB Server.
2. Configure an `Application` object for DB Server.
3. Install DB Server on its host.
4. Set up any environment variables that are specific to your RDBMS type.

For detailed instructions, see the *Framework 7.6 DB Server User's Guide*.

End of procedure

Next Steps

- Deploying Interaction Concentrator.

Deploying Interaction Concentrator

Deploy as many instances of Interaction Concentrator as you need. The following summary procedure consists of major steps, each of which is described as a separate procedure in this section.

Procedure: Deploying Interaction Concentrator

Purpose: To configure and install as many instances of Interaction Concentrator as needed on either a UNIX or Windows operating system.

Prerequisites

- Deploying DB Server

Start of procedure

1. Import the application template for Interaction Concentrator (see page 43).
2. Configure an `Application` object for Interaction Concentrator (see page 43).
3. Install Interaction Concentrator on its host (see page 46).

End of procedure

Next Steps

- Importing the application template

Note: Interaction Concentrator uses the `Call Concentrator` application type for its `Application` object in the Configuration Layer.

Environment Assumptions

The instructions in this section assume that you are creating new `Application` objects under the `Environment` folder in Configuration Manager, in either a single-tenant or multi-tenant configuration environment. To create `Application` objects under a particular `Tenant` folder in a multi-tenant configuration environment, replace the word `Environment` with the name of your `Tenant` folder in the configuration instructions.

Importing the Application Template

Before you can configure an `Application` object for Interaction Concentrator, you must import its application template. The application template provides a majority of the configuration options, as well as the default values for them.

Procedure:
Importing the application template

Purpose: To create an application template that you can use to create as many `Application` objects of the same type as you need.

Start of procedure

1. Open the Configuration Manager main window.
2. Select the `Environment > Application Templates` folder.
3. From the `File` menu, select `Import Application Template`.
4. In the `Look In` box, click the down arrow.
5. Locate the Interaction Concentrator 7.6 product CD, and open the `TEMPLATES` folder.
6. Select the template file for Interaction Concentrator; it is called `Interaction_Concentrator_760.apd`.
7. Click `Open` to open the `Properties` dialog box for the template.
8. Make any changes that you require.
9. Click `OK` to save the template and close the `Properties` dialog box.

End of procedure

Next Steps

- Configuring an ICON application object

Configuring an ICON Application Object

After you import the application template, you can create and configure an `Application` object for your Interaction Concentrator by using Configuration Manager.

Procedure:
Configuring an ICON application object

Prerequisites

- Importing the application template

Start of procedure

1. Open the Configuration Manager main window.
2. Select the `Environment > Applications` folder.
3. From the `File` menu, select `New > Application`.
4. From the available application templates in the `Browse` dialog box, select the template that you imported for Interaction Concentrator.
5. In the `Properties` dialog box, click the `General` tab, and then enter a name for this application. The application template provides information about the application type and version.

> **Note:** Interaction Concentrator uses the `Call Concentrator` application type for its `Application` object in the Configuration Layer.

6. Click the `Server Info` tab, and then specify the following properties:
   - `Host`—Enter the name or IP address of the computer on which you want to install and/or run this server.
   - `Communication Port`—Enter any numeric port value in the `1–65,535` range. Although ICON ignores this property, the Configuration Manager interface requires a value for this property.
7. Click the `Start Info` tab, and then specify the following properties:
   - `Working Directory`—Enter the full path to the directory from which the application starts.
   - `Command Line`—Enter the command line that is used to start the application.
   - `Command Line Arguments`—Enter any additional command-line parameters that are used to start the application. For information about command-line parameters, see “Command-Line Parameters” on page 58.

> **Note:** These properties are updated automatically during the installation procedure.

8. Click the `Options` tab, and then specify/change the values of the configuration options. For information about ICON configuration options, see “ICON Configuration Options” on page 67.

   Also:
   - Configure both ICON-specific log options and common log options in the log-related configuration sections. For option descriptions, see “log Section” on page 100 and the *Framework 7.6 Configuration Options Reference Manual*.
   - If Interaction Concentrator’s working directory differs from the directory to which the application is installed, configure an option named `messagefile` in the `log` section. As the value of this option,

specify the full path to the application-specific log messages file (`icon.lms`). Otherwise, ICON will be unable to generate its specific log events.

- Configure an HTTP listener by creating a `listeners` section and specifying the appropriate option. For option descriptions, see "listeners Section" on page 99.
- If you have already deployed another ICON instance, make sure that only one of the ICON applications is configured to store configuration data and the history of configuration changes. For all other instances, deactivate the configuration data storage by configuring one of the following values for the `role` option in the `callconcentrator` section:
  - Explicitly specify one or more values other than `cfg`.
  - Exclude the `cfg` value by using the tilde symbol (`~`) (that is, set the `role` option to `~cfg`).

  For more information about the `role` option, see the option description on page 76.

9. The `Tenants` tab is displayed only in a multi-tenant environment. Click the `Tenants` tab, and then click `Add` to add all tenants that this ICON application will serve.

Notes: It is important to add all tenants from whose resources (switches, DNs, and agents) ICON will collect data.

If this ICON instance is required to monitor the objects that are configured under the `Environment` folder, assign the `Environment` tenant among the other tenants.

10. Click the `Connections` tab, and then do the following:

- If this ICON instance is configured to process CTI–related data, add a connection to T-Server. For information about how to enable Advanced Disconnect Detection Protocol (ADDP) for this connection, see the *Framework 7.6 Deployment Guide*.

Note: If you have a simple multi-site topology that includes one ICON instance and multiple T-Servers, add a connection to each T-Server on the `Connections` tab. Each `T-Server Application` object must have a `Switch` object assigned to it.

- If this ICON instance is configured to process outbound data in an environment with Genesys Outbound Contact, add a connection to one or more OCS applications. For information about how to enable ADDP for these connections, see the *Framework 7.6 Deployment Guide*.

- To enable ADDP between ICON and Configuration Server, add the Configuration Server application (named `confserv`) to the `Connections` tab, and then specify the values for the connection protocol, in seconds. For more information, see the *Framework 7.6 Configuration Manager Help*.
- If you installed the Management Layer, add a connection to Message Server, in order to provide alarm signaling and centralized logging capabilities.

Note: You can add a connection to Message Server for all or a set of `Application` objects after you configure them. To launch a wizard that configures connections for multiple `Application` objects, select two or more `Application` objects, right-click, and then select `Manage Connections` from the shortcut menu. For more information, see the *Framework 7.6 Configuration Manager Help*.

- After you configure one or more `DAP Application` objects (see page 54), add to the ICON `Connections` tab any `DAP Application` objects through which this ICON instance will access IDBs.

11. Click `OK` to save your changes and close the `Properties` dialog box.

End of procedure

Next Steps

- Manually install the Interaction Concentrator application on your specific operating system. Do one of the following:
  - Installing the ICON application on UNIX, page 46
  - Installing the ICON application on Windows, page 47

Installing the ICON Application

After you use Configuration Manager to create an `Application` object for Interaction Concentrator, install Interaction Concentrator. The following procedures provide instructions for installing the application on UNIX and Windows operating systems, respectively.

Procedure: Installing the ICON application on UNIX

Purpose: To install the ICON application by using the shell script provided on the Interaction Concentrator product CD.

Prerequisites

- Configuring an ICON application object

Start of procedure

1. On the Interaction Concentrator 7.6 product CD, in the appropriate `icon/<operating_system>/` directory, locate the `install.sh` shell script.
2. Run this script from the command line by entering the following command:

   ```
   install.sh
   ```
3. When prompted, specify the host name of the computer on which you want to install ICON.
4. When prompted, specify:
   a. The host name of the computer on which Configuration Server is running.
   b. The port client that applications use to connect to Configuration Server.
   c. The user name that is used to log in to the Configuration Layer.
   d. The password that is used to log in to the Configuration Layer.
5. The installation displays the list of `Application` objects of `Call Concentrator` type that are configured for this host. Type the number corresponding to the ICON `Application` that you want to install.
6. Specify the destination directory into which you want to install ICON.
7. If prompted for the version of the product to install, (32- or 64-bit), select the appropriate version for your environment.

As soon as the installation process is complete, a message appears, announcing that installation was successful. The installation process creates a directory with the name that you specified in Step 6, and it places ICON in this directory.

End of procedure

Next Steps

- Use the database scripts to initialize IDB. See “Creating IDB” on page 49.

Procedure:
Installing the ICON application on Windows

Purpose: To install the ICON application by using the Genesys installation wizard provided on the Interaction Concentrator product CD.

Start of procedure

1. On the Interaction Concentrator 7.6 product CD, open the `icon\Windows\` directory.
2. Locate and double-click `setup.exe` to start the Genesys Installation Wizard.
3. On the `Welcome` page, click the `About` button to review the `read_me` file for this installation package. The file also contains a link to the Release Notes file for Interaction Concentrator.
4. Click `Next` to proceed with the installation.

> **Note:** Click `Next` at the end of each step to proceed to the next page in the wizard.

5. On the `Connection Parameters to the Genesys Configuration Server` page, specify the following login parameters:
   - Host and port of Configuration Server
   - User name and password used to log in to the Configuration Layer
6. The `Select Application` page displays all applications of the `Call Concentrator` type in the Configuration Database. When you select one application from the list, the wizard displays some of the parameters that are configured for that application (in particular, the application type, host, working directory, command line, and command-line arguments).

   Select the application that you want to install.
7. The `Choose Destination Location` page displays the destination directory, as specified in the `Working Directory` property of the ICON `Application` object (see Step 7 on page 44). If the path that is configured as `Working Directory` is invalid, the wizard generates the following path to the destination directory:

   `C:\Program Files\GCTI\Interaction Concentrator\<ICON Application Name>\`

   If necessary, click one of the following:
   - `Browse`—To select another destination folder. The wizard updates the `Application` object's `Working Directory` property in the Configuration Database.
   - `Default`—To reinstate the path that is specified in the `Application` object's `Working Directory` property.
8. On the `Ready to Install` information page, click one of the following:
   - `Back`—To update any installation information.
   - `Install`—To proceed with installation. The `Installation Status` window appears, showing the installation progress.
9. On the `Installation Complete` page, click `Finish`.

End of procedure

As a result of the installation, the wizard adds `Interaction Concentrator` icons to the following:

- Windows `Start` menu, under `Programs > Genesys Solutions`.
- Windows `Add or Remove Programs` dialog box, as a Genesys server.
- Windows `Services` list, as a Genesys service with the `Automatic` startup type.

The wizard also places the following files into the folder to which you installed Interaction Concentrator:

- A template file for the ICON attached data specification. (For more information about the attached data specification file, see the "Attached Data Specification File" section of the *Interaction Concentrator 7.6 User's Guide.*)
- An example of the ICON attached data specification.
- Scripts for IDB, for every supported RDBMS type.

Next Steps

- Use the database scripts to initialize IDB. See "Creating IDB".

Creating IDB

You can use any of the supported RDBMSs to host your IDB. Ask your Database Administrator to create a new database for each IDB instance that you intend to deploy for ICON data storage. Then, initialize each IDB instance, using the instructions in this section.

Note: The user account that is created for IDB must have permissions to create database objects such as tables, stored procedures, and sequences.

Initialization Scripts

After you install the ICON application, the `scripts` subfolder in the directory to which you installed ICON contains a set of 26 initialization and 2 sample scripts for each RDBMS type. Table 3 describes the purpose of these scripts. In the script names in Table 3, `<db_type>` is a placeholder for the specific RDBMS type (`db2`, `mssql`, or `ora` [for Oracle]).

Note: If you are migrating from an existing IDB, do not simply apply all the scripts in Table 3. To avoid damaging or erasing existing data, follow the migration procedures that are described in the Interaction Concentrator chapters in the *Genesys 7 Migration Guide*.

Table 3: IDB Initialization Scripts

Script Name	Description
01_gcc_<db_type>_schema.sql	Initializes the principal IDB schema.
02_gos_<db_type>_schema.sql	Initializes the outbound schema in IDB.
03_gcc_<db_type>_stp_api.sql	Creates general stored procedures.
04_gcc_<db_type>_schema_2.sql	Upgrades the IDB schema to include the `G_VIRTUAL_QUEUE` table.
05_gcc_<db_type>_stp_api_2.sql	Creates stored procedures that associate calls with virtual queues.
06_gcc_<db_type>_schema_3.sql	Upgrades the IDB schema to include custom-state tables.
07_gcc_<db_type>_idx_change_3.sql	Creates indexes for various IDB tables.
08_mcr_<db_type>_schema.sql	Upgrades the IDB schema to include the Multimedia user data tables.
09_gcc_<db_type>_stp_api_3.sql	Creates stored procedures related to call control, login sessions, and user data.
10_gcc_<db_type>_stp_sys_3.sql	Creates special (service) stored procedures.
11_gcc_<db_type>_idx_change_5.sql	Creates indexes for various IDB tables.
12_gcc_<db_type>_stp_api_4.sql	Upgrades stored procedures for configuration data tracking.
13_gcc_<db_type>_svc_change_5.sql	Changes the service stored procedures.
14_gcc_<db_type>_stp_api_5.sql	Upgrades stored procedures for Open Media data tracking.
15_gcc_<db_type>_stp_sys_5.sql	Upgrades special (service) stored procedures.
16_gcc_<db_type>_stp_api_6.sql	Updates stored procedures for user data tracking.
17_gcc_<db_type>_idx_change_6.sql	Creates index for `G_PARTY` IDB table.
18_gcc_<db_type>_stp_api_7.sql	Creates stored procedures related to call control, configuration data and user data tracking.
19_gcc_<db_type>_stp_sys_7.sql	Upgrades merge procedures.
20_gcc_<db_type>_disp.sql	Creates stored procedures for event dispatcher functionality.

Table 3: IDB Initialization Scripts (Continued)

Script Name	Description
21_gcc_<db_type>_dict.sql	Populates the dictionary tables with metadata.
22_gcc_<db_type>_version.sql	Updates the version of the IDB schema.
purge_gcc_<db_type>_proc.sql	Creates service table and stored procedures for the new release 7.6.1 purge IDB procedure (gsysPurge76).
gcc_<db_type>_clean_init.sql (optional)	Creates tables and indexes in IDB to clean up (purge) the database.
gcc_<db_type>_clean_upgrade.sql (optional)	Updates tables and indexes in IDB to clean up (purge) the database.
gcc_<db_type>_clean_api.sql (optional)	Creates stored procedures to purge IDB.
sample_gcc_<db_type>_custdisp_api.sql	Serves as a sample script, illustrating how to modify the custom dispatcher stored procedures.
sample_gcc_<db_type>_custdisp_schema.sql	Serves as a sample script, illustrating how to create a custom attached data storage table.

Initializing the Database

For a first-time initialization of IDB, run the provided scripts in sequential order, starting with `01_gcc_<db-type>_schema.sql` and ending with `22_gcc_<db_type>_version.sql`. Follow the RDBMS-specific instructions in the following procedures.

Notes: The IDB initialization scripts create default (empty) custom dispatchers without first dropping any existing stored procedures named `gudCustDisp1` and `gudCustDisp2`. This is to decrease the risk of overwriting customer-created stored procedures. However, if the `gudCustDisp1` and `gudCustDisp2` custom dispatcher stored procedures already exist in IDB, script `09_gcc_<db_type>_stp_api_3.sql` will return an error, which you can ignore.

Genesys provides the `sample_gcc_<db_type>_custdisp_api.sql` and `sample_gcc_<db_type>_custdisp_schema.sql` scripts to help you understand how you can modify the stored procedures for customized attached data processing. Do not execute the sample scripts during installation. For more information about configuring your ICON application to support customized attached data processing, see the chapter about customized attached data processing in the *Interaction Concentrator 7.6 User's Guide.*

Procedure: Initializing IDB on DB2

Purpose: To initialize IDB by running the initialization scripts provided for a DB2 database.

Start of procedure

1. Go to the directory to which you installed ICON.
2. Go to the `scripts\db2` subdirectory.

Execute every script in sequential order, starting with `01_gcc_db2_schema.sql` and ending either with `22_gcc_db2_version.sql` or, if you want to implement purge functionality, with `purge_gcc_<db_type>_proc.sql`, or `gcc_db2_clean_init.sql` and `gcc_db2_clean_api.sql`. To execute the scripts:

a. Insert the following command line at the beginning of each script, providing appropriate values for the placeholders:

```
connect to <dbname> user <user> using <password>@
```

b. Use the following command line to load each initialization script:

```
db2 +w -td@ -f<script_name>
```

End of procedure

Next Steps

- Configuring a DAP

Procedure: Initializing IDB on Microsoft SQL

Purpose: To initialize IDB by running the initialization scripts provided for a Microsoft SQL database.

Start of procedure

1. Go to the directory to which you installed ICON.
2. Go to the `scripts\mssql` subdirectory.
3. Execute every script in sequential order, starting with `01_gcc_mssql_schema.sql` and ending either with `22_gcc_mssql_version.sql` or, if you want to implement purge functionality, with `purge_gcc_<db_type>_proc.sql`, or with `gcc_db2_clean_init.sql` and `gcc_db2_clean_api.sql`.

To execute the scripts, use the following command line to load each initialization script, providing appropriate values for the placeholders:

```
isql -S <dbms_server> -d <dbname> -U <user> -P <password> -i
<script_name>
```

End of procedure

Next Steps

- Configuring a DAP

Procedure: Initializing IDB on Oracle

Purpose: To initialize IDB by running the initialization scripts provided for an Oracle database.

Start of procedure

1. Go to the directory to which you installed ICON.
2. Go to the `scripts\oracle` subdirectory.
3. Execute every script in sequential order, starting with `01_gcc_ora_schema.sql` and ending either with `22_gcc_ora_version.sql` or, if you want to implement purge functionality, with `purge_gcc_<db_type>_proc.sql`, or with `gcc_db2_clean_init.sql` and `gcc_db2_clean_api.sql`. To execute the scripts, log in to the `sqlplus` command processor, and type the following at the command prompt:

   ```
   @ <script_name>
   ```

End of procedure

Next Steps

- Configuring a DAP

Configuring a DAP

If you are unsure how to use Configuration Manager to create a new `DAP Application` object, refer to one of the following sources:

- Appendix A, “Standard Configuration Procedure” of the *Framework 7.6 Deployment Guide.*
- *Framework 7.6 DB Server User’s Guide.*

In addition to the standard configuration steps, complete the following procedure.

Procedure: Configuring a DAP

Purpose: To configure a `DAP Application` object for IDB.

Start of procedure

1. On the `General` tab, when you specify the application name, keep in mind that the DAP can have the same name as the database itself. However, if you are using multiple access points to the same database, make their names unique.
2. On the `General` tab, click `Browse` to locate the DB Server through which this database is to be accessed. This must be the DB Server that is either deployed or reused for ICON purposes (see Step 3 on page 40 and "Deploying DB Server" on page 41).

   Note: Do not select the `JDBC Connection` check box, because it does not apply to database connections through DB Server.

3. On the `DB Info` tab, specify the properties as follows:
   - `DBMS Name`—The name or alias that identifies the RDBMS that handles IDB. The value of this option is communicated to DB Server so that it connects to the correct RDBMS:
     - For Oracle, set the value to the name of the Listener service (also known as a *database alias*).
     - For Microsoft SQL, set the value to the name of the SQL server (usually the same as the host name of the computer on which Microsoft SQL runs).
     - For DB2, set the value to the name or alias name of the database, as specified in the db2 client configuration.
   - `DBMS Type`—The type of RDBMS that handles IDB. You must set a value for this property.
   - `Database Name`—The name of the IDB instance to be accessed, as it is specified in the RDBMS that handles this database. You must set a value for this property, unless you specify `oracle` or `db2` for `DBMS Type`. For Informix and Microsoft SQL, the value is the name of the database to which the client will connect.
   - `User Name`—The user name for accessing IDB, as established in the SQL server. You must set a value for this property.
   - `Password`—The password for accessing IDB, as established in the SQL server.

- `Re-enter Password`—Confirmation for the value that you entered for `Password`.
- `Case Conversion`—The case conversion method for key names of key-value lists that come from DB Server. This value specifies whether, and how, a client application converts the field names of a database table when it receives data from DB Server:
  - `upper`—Field names are converted into uppercase.
  - `lower`—Field names are converted into lowercase.
  - `any`—Field names are not converted.

  This option does not affect the values of key-value lists that come from DB Server—the actual data is presented exactly as it appears in the database tables.

Warning! For the `Case Conversion` option, use the default value (`any`), unless Genesys Technical Support directs you to do otherwise.

Note: When configuring a `DAP Application` object for IDB, do not configure any properties on the `JDBC Info` tab.

4. If you intend to use multiple database access points to write different types of ICON data to different databases, specify which type(s) of data this particular database access point must handle. On the `Options` tab:
   a. Create a section named `callconcentrator`.
   b. Within the `callconcentrator` section, create a configuration option named `role`.
   c. Set the option value to indicate the types of data that will be stored through this DAP. For more information, see the option description on page 116.

      For optimal performance, Genesys recommends the following sets of values for a given database access point:
      - `gcc, gud, gls`
      - `cfg`
      - `gos`

End of procedure

Next Steps

- After you configure a `DAP Application` object, add it to the `Connections` tab of the ICON application that will use this DAP as an interface to IDB.

Chapter

4 Starting and Stopping Interaction Concentrator

This chapter describes the prerequisites for Interaction Concentrator startup, and it provides instructions for starting and stopping Interaction Concentrator (ICON). It contains the following sections:

Overview

You can start and shut down Interaction Concentrator components by using the Management Layer, a startup file, a manual procedure, or Services Manager.

All of these methods usually require command-line parameters for a server application as well as an executable file name. The next section describes the command-line parameters that are common to most Genesys server applications. Subsequent sections describe the startup and shutdown procedures.

Note: For information about using the Management Layer, startup files, and Services Manager for startup, see the *Framework 7.6 Deployment Guide.*

Command-Line Parameters

The following startup command-line parameters are supported by Interaction Concentrator:

`-host`	The name of the host on which Configuration Server is running.
`-port`	The communication port that client applications must use to connect to Configuration Server.
`-app`	The exact name of an application as configured in the Configuration Database.
`-V`	The version of a component. Note that specifying this parameter does not start an application, but instead returns its version number. You can use either an uppercase letter (`V`) or lowercase letter (`v`).
`-lmspath`	The full path to the log messages files that an application uses to generate log events. (These files are the common file named `common.lms` and the application-specific file with the extension `*.lms`.) Use this parameter when the common and application-specific log message files are located in a directory other than the application's working directory—for example, when the application's working directory differs from the directory to which the application was originally installed. Note that if the full path to the executable file is specified in the startup command line (for instance, `c:\gcti\multiserver.exe`), the path that is specified for the executable file is used to locate the `*.lms` files, and the value of the `lmspath` parameter is ignored.

Warning! An application that does not locate its `*.lms` file at startup cannot generate application-specific log events and send them to Message Server.

Starting ICON

This section provides manual startup instructions for ICON server. You can start ICON in any of the following ways:

- From SCI (see page 59).
- Manually on UNIX (see page 60).
- Manually on Windows (see page 61).
- As a Windows Service on Windows (see page 62).

Starting ICON with Solution Control Interface

Complete the following procedure to start ICON with Solution Control Interface (SCI).

Procedure: Starting ICON with SCI

Prerequisites

Genesys recommends that the following applications be running before you start ICON:

- The DB Server that provides access to IDB.
- The relational database management system.
- T-Server.
- Outbound Contact Server, if ICON is configured to collect data from OCS.
- Interaction Server, if ICON is configured to collect data from Multimedia.

If you have configured ICON to store attached data, ensure that there is a proper attached data specification file in ICON's working directory. (By default, ICON uses the `ccon_adata_spec.xml` file.)

For a short period of time after starting or restarting, ICON may produce `[cp:...]` or FSM errors in the log. These errors occur when ICON encounters elements of interactions that it cannot resolve because the interactions were already in progress when ICON was started or restarted. You can safely ignore these errors.

For detailed instructions about starting the Genesys components on which Interaction Concentrator depends, see:

- *Framework 7.6 Deployment Guide*
- *Framework 7.6 T-Server Deployment Guide* for your particular T-Server type
- *Framework 7.6 DB Server User's Guide*
- *Outbound Contact 7.6 Deployment Guide*
- *Multimedia 7.6 Deployment Guide*

Start of procedure

1. On the list pane in the SCI `Applications` view, select your ICON `Application` object.
2. Do one of the following:
   - On the toolbar, click the `Start` button.
   - From the `Action` menu, select `Start`.

- Right-click the Application object to access the shortcut menu, and then select Start.

3. In the confirmation box that appears, click Yes.

   SCI starts your Interaction Concentrator application.

End of procedure

Next Steps

- You have completed all the steps necessary to start ICON using SCI.

Starting ICON Manually

Complete the following procedure to start ICON manually on UNIX.

Procedure: Starting ICON manually on UNIX

Start of procedure

1. Go to the directory to which you have installed ICON.
2. Enter the name of the ICON executable, followed by the appropriate command-line parameters, using the following syntax:

   ```
   ./icon -host <hostname> -port <portno> -app <application>
   ```

   Where:

   - *hostname* is the name of the host on which Configuration Server is running.
   - *portno* is the communication port that client applications must use to connect to Configuration Server.
   - *application* is the name of the Interaction Concentrator Application object, as defined to Configuration Server.

Note: If the host name or application name contains spaces or hyphens (-), enclose them in double quotation marks.

For example, to start ICON with command-line parameters that specify the host as cs-host, the port as 2020, and the name as ICON 03, enter the following:

```
./icon -host "cs-host" -port 2020 -app "ICON 03"
```

End of procedure

Next Steps

- You have completed all the steps necessary to start ICON manually on UNIX.

Starting ICON on Windows

Complete the following procedure to start ICON on Windows.

Procedure: Starting ICON on Windows

Purpose: To start ICON from the `Start > Programs` menu, or from the console window.

Start of procedure

1. Open a console window.
2. Go to the directory to which you installed Interaction Concentrator.
3. Enter the following command line:

   `icon.exe -host <hostname> -port <portno> -app <application>`

   Where:
   - *`hostname`* is the name of the host on which Configuration Server is running.
   - *`portno`* is the communication port that client applications must use to connect to Configuration Server.
   - *`application`* is the name of the Interaction Concentrator `Application` object, as defined to Configuration Server.

Note: If the host name or application name contains spaces or hyphens (-), enclose them in double quotation marks.

For example, to start ICON with command-line parameters that specify the host as `cs-host`, the port as `2020`, and the name as `ICON 03`, enter the following:

```
icon.exe -host "cs-host" -port 2020 -app "ICON 03"
```

End of procedure

Next Steps

- You have completed all the steps necessary to start ICON on Windows.

Starting ICON as a Windows Service

On Microsoft Windows platforms, by default, the installation process installs Interaction Concentrator as a Windows Service. If you stopped ICON from running as a Windows Service and need to start it again as a Windows Service, complete the following procedure.

Procedure: Starting ICON as a Windows service

Start of procedure

1. Open the Windows `Control Panel`, and then double-click the `Services` icon. The `Services` dialog box opens.
2. In the `Services` list box, select your ICON service, and then click `Start`. (If you disabled Interaction Concentrator from operating as a Windows Service, the `Start` option for this application will not be available.)

Note: You can install the Local Control Agent (LCA) as a Windows Service with the user interface disabled. In this case, all servers that are started through SCI are started without a console, unless you specifically select the `Allow Service to Interact with Desktop` check box for both LCA and ICON.

End of procedure

Next Steps

- You have completed all the steps necessary to start ICON as a Windows service.

Stopping ICON

You can stop ICON in any of the following ways:

- From SCI (see page 63). (This is the recommended method.)
- Manually on UNIX (see page 63).
- Manually on Windows (see page 64).
- As a Windows Service on Windows (see page 65).

Note: To prevent ICON from self-starting, make sure that you clear the `autorestart` property in the ICON `Application` object in Configuration Manager.

Stopping ICON with Solution Control Interface

If you are using LCA and SCS, complete the following procedure to stop ICON with SCI.

Procedure: Stopping ICON using SCI

Start of procedure

1. On the list pane in the SCI `Applications` view, select your ICON `Application` object.
2. Do one of the following:
   - On the toolbar, click `Stop`.
   - From the `Action` menu, select `Stop`.
   - Right-click the `Application` object to access the shortcut menu, and then select `Stop`.
3. In the confirmation box that appears, click `Yes`.

   SCI stops your Interaction Concentrator application.

End of procedure

Next Steps

- You have completed all the steps to stop ICON using SCI.

Stopping ICON on UNIX

Stop ICON on UNIX by using one of the following procedures.

Procedure: Stopping ICON on UNIX from the command line

Start of procedure

- On the command line, enter the following:

  `kill -SIGTERM <processid>`

  Where `<processid>` is the application's UNIX process ID.

End of procedure

Next Steps

- You have completed all the steps to stop ICON from the command line.

Procedure: Stopping ICON on UNIX from the console window

Start of procedure

- From the active console window, press `CTRL+C`.

End of procedure

Next Steps

- You have completed all the steps to stop ICON from the console window.

Note: If you are using LCA and SCS, you can also use SCI to stop ICON (see “Stopping ICON with Solution Control Interface” on page 63).

Stopping ICON on Windows

If ICON is running as an application—not as a Windows Service—stop it using the following procedure.

Procedure: Stopping ICON on Windows from the console window

Start of procedure

- From the application’s console window, press `CTRL+C`.

End of procedure

Next Steps

- You have completed all the steps to stop ICON from the console window.

Note: If you are running ICON as a Windows Service, you should stop it only from the Services Control Manager (see “Stopping ICON as a Windows Service”).

Stopping ICON as a Windows Service

To stop Interaction Concentrator running as a Windows Service, use the following procedure.

Procedure: Stopping ICON running as a Windows service

Start of procedure

1. Open the `Control Panel`, and then double-click the `Services` icon. The `Services` dialog box opens.
2. In the `Services` list box, select your ICON service, and then click `Stop`.

End of procedure

Next Steps

- You have completed all the steps to stop ICON running as a Windows Service.

Chapter

5 Configuration Options

This chapter describes the configuration options that you can set for effective operation of your Interaction Concentrator (ICON) `Application` object. It contains the following sections, one for each `Application` object for which you must set options:

Note: In addition to the configuration options described in this chapter, Interaction Concentrator supports the *common log options* that are described in the *Framework 7.6 Configuration Options Reference Manual.*

Interaction Concentrator also connects to Message Server, but no ICON-specific configuration settings are required. For information about Message Server configuration options, see the *Framework 7.6 Configuration Options Reference Manual*.

ICON Configuration Options

This section describes the options that you define on the `Options` tab of your Interaction Concentrator (ICON) `Application` object.

Available Options

You do not have to configure any options in order to start Interaction Concentrator. However, the available configuration options can greatly affect the performance and effectiveness of your Interaction Concentrator

application. Review the information about all of the configuration options for the ICON `Application` object, in order to identify and configure the settings that are appropriate to your environment.

Configuration Server recognizes the following sections for ICON `Application` objects:

- `callconcentrator` (see page 68)
- `custom-states` (see page 88)
- `dbw-error-reactions` (see page 91)
- `filter-data` (see page 93)
- `listeners` (see page 99)
- user-defined section identifying the HTTP connection (see page 100)
- `log` (see page 100)

callconcentrator Section

You must name this section `callconcentrator` in the configuration. The options in this section configure the main functionality of Interaction Concentrator.

Note: The name of this configuration option section is derived from the type of application that is used for the Interaction Concentrator `Application` object in the Genesys Configuration Layer. Otherwise, there is no relationship between the Genesys Interaction Concentrator and Call Concentrator products.

Table 4 describes the options in the `callconcentrator` section. For ease of reference, the options are grouped by area of functionality (presented in alphabetical order) as follows:

- Agent metrics, page 69
- Attached data, page 71
- Custom dispatcher, page 74
- Database writing, page 74
- ICON Role, page 76
- IDB, page 77
- In-memory queue, page 77
- Multimedia, page 78
- Open Media, page 79
- Operational memory, page 79
- Outbound metrics, page 81
- Persistent queue, page 82
- Scenario recognition, page 85
- Synchronization, page 85

- Virtual Queue, page 87

Table 4: ICON Configuration Options—callconcentrator Section

Area of Functionality	Option	Description
Agent metrics	gls-acw-first	Specifies which interaction ICON associates with after-call work (ACW). By default, ICON associates after-call work metrics with the voice interaction that immediately precedes the completion of the after-call work (the last voice interaction). Setting this option to `true` enables ICON to associate after-call work with the voice interaction that immediately precedes the *start* of the after-call work (the first voice interaction). In this case, subsequent voice interactions that occur during the period of after-call work are considered as related to ACW processing and should not interrupt measurement of ACW-related metrics. When the agent logs out, changes his or her state to `Ready`, or goes `NotReady` for any reason other than to perform after-call work, ICON reports the end of the current ACW state. This option applies to all switches that ICON is configured to monitor; but, this option's value does not override the value of the `gls-acw-first` configuration option (described on page 106) if configured within the switch's configuration object. Default Value: `false` Valid Values: `false` ICON associates the last voice interaction with after-call work. `true` ICON associates the first voice interaction with after-call work. Changes take effect: After restart
	gls-stats-update	Specifies whether agent metrics (such as the duration of a particular agent state) are updated in IDB (`GS_AGENT_STAT` and `GS_AGENT_STAT_WM` tables) as the agent login session progresses. By default (value = `false`), ICON stores agent metrics only after an agent login session ends. Default Value: `false` Valid Values: `true` Agent metrics (such as a state duration) are updated dynamically in IDB. `false` Agent metrics (such as a state duration) are stored in IDB after a login session ends. Changes Take Effect: After restart

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
Agent metrics (continued)	gls-stats-update-delta	Specifies the minimum change, in seconds, in the duration of an agent state that causes an updated metric's value to be stored in IDB. ICON processes this option only if you set the `gls-stats-update` option to `true`. Default Value: `10` Valid Values: Any integer Changes Take Effect: After restart
	gls-store-event-seq	Specifies whether ICON stores event sequence numbers when events related to an agent login session trigger creation of new records in the following IDB tables: • `G_AGENT_STATE_HISTORY` • `G_AGENT_STATE_RC` • `G_DND_HISTORY` By default, ICON retrieves event sequence numbers from T-Server or Interaction Server events and stores the numbers along with new records in the above tables. **Note:** To provide event sequence numbers with multimedia events, Interaction Server release 7.6 is required. Default Value: `1` Valid Values: `0` ICON does not store a sequence number of the event that triggered a new record in an agent-related table. `1` ICON stores a sequence number of the event that triggered a new record in an agent-related table. Changes Take Effect: After restart

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
Attached data	adata-default-storage	Specifies the default destination for storing attached data, for a key that is not included in the XML specification file specified by the `adata-spec-name` option value. ICON processes this option only if you enable attached data storage by setting the `role` option to either `all` or `gud` (see page 76). This option applies to voice and Multimedia interactions. Default Value: `public` Valid Values: `public, secure` Changes Take Effect: After restart **Note:** For descriptions of these values, see the storage types described in the chapter about attached data in the *Interaction Concentrator 7.6 User's Guide.*
	adata-extensions-history	Specifies what changes to a key's value must be recorded in IDB, for a key that originates from the `Extensions` TEvent attribute, but that is not included in the XML specification file specified by the `adata-spec-name` option value. ICON processes this option only if you enable attached data storage by setting the `role` configuration option to either `all` or `gud` (see page 76). This option applies to voice and Multimedia interactions. Default Value: `none` Valid Values: `none` No value for a given key is recorded in IDB. `first` Only the first value for a given key is recorded in IDB. `last` Only the last value for a given key is recorded in IDB. `all` Every change in value for a given key is recorded in IDB. Changes Take Effect: After restart

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
Attached data (continued)	adata-reasons-history	Specifies what changes to a key's value must be recorded in IDB, for a key that originates from the `Reasons` TEvent attribute, but that is not included in the XML specification file specified by the `adata-spec-name` option value. ICON processes this option only if you enable attached data storage by setting the `role` option to either `all` or `gud` (see page 76). This option applies to voice interactions only. Default Value: `none` Valid Values: `none` No value for a given key is recorded in IDB. `first` Only the first value for a given key is recorded in IDB. `last` Only the last value for a given key is recorded in IDB. `all` Every change in value for a given key is recorded in IDB. Changes Take Effect: After restart
	adata-spec-name	Specifies the name of the XML file that contains the specification of attached data. The specification lists attached keys. For every key, the specification also includes: • Key source—The TEvent or Multimedia Interaction Reporting event attribute, such as `UserData`, `Reasons`, or `Extensions`, from which the key originated. • History type—The changes to the key's value that must be recorded (for example, all changes, only the first change, or only the last change). ICON processes this option only if you enable attached data storage by setting the `role` option to either `all` or `gud` (see page 76). For more information about the specification, see the section about the attached data specification file in the *Interaction Concentrator 7.6 User's Guide.* Default Value: `ccon_adata_spec.xml` Valid Values: Any valid name Changes Take Effect: After restart

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
Attached data (continued)	adata-userdata-history	Specifies what changes to a key's value must be recorded in IDB, for a key that originates from the `UserData` reporting event attribute, but that is not included in the XML specification file specified by the `adata-spec-name` option value. ICON processes this option only if you enable attached data storage by setting the `role` option to either `all` or `gud` (see page 76). This option applies to voice and Multimedia interactions. Default Value: `none` Valid Values: `none` No value for a given key is recorded in IDB. `first` Only the first value for a given key is recorded in IDB. `last` Only the last value for a given key is recorded in IDB. `all` Every change in value for a given key is recorded in IDB. Changes Take Effect: After restart
	suppress-user-data	Specifies whether ICON instructs T-Server to propagate attached data only when the attached data changes. This optimizes processing of attached data by reducing network traffic. **Note:** This option can be set at the level of the ICON application or the switch (see the description of the `suppress-user-data` option on page 112). ICON automatically detects the switch-level option setting. If the switch-level option is set to the (default) value of `1` (unchanged attached data suppressed), T-Server TEvents are optimized for all ICON applications that connect to the T-Servers for that switch. In this case, the switch-level option setting overrides any application-level settings of `0` (unchanged attached data not suppressed). If the switch-level option is set to `0`, an application-level setting of `1` will override it. Default Value: `1` Valid Values: `0` Unchanged attached user data is not suppressed. `1` Unchanged attached user data is suppressed. Changes Take Effect: After restart

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
Custom dispatcher	gud-cust-disp	Specifies whether ICON calls a custom stored procedure to handle attached data and store the information in custom tables. **Note:** ICON starts executing the new custom dispatcher as soon as the new configuration option value is set. Processing that was begun by the old custom dispatcher of interaction information stored in the persistent queue is handled in IDB by the old custom dispatcher. Default Value: `0` Valid Values: `0` ICON does not call a custom dispatcher. `1` ICON calls the `gudCustDisp1` stored procedure. `2` ICON calls the `gudCustDisp2` stored procedure. Changes Take Effect: Immediately **Note:** For more information, refer to the section about custom dispatchers in the *Interaction Concentrator 7.6 User's Guide.*
	gud-cust-disp-groups	Specifies the maximum number of key groups that ICON can process. If you code more than the maximum number of groups in the XML file, ICON ignores the extra key groups and does not provide data to the active custom dispatcher. Key names that you specify must be unique both within and across key groups. The maximum number of keys that you can specify for any particular key group is limited to 34 (17 key-value pairs for string values, and 17 for integer values). Default value: `16` Valid values: `0`–`255` (`0` indicates that ICON will process no group) Changes Take Effect: After restart
Database writing	dbw-request-tout	Specifies the amount of time, in seconds, that ICON waits for the completion of a database writing transaction. If a transaction is not completed when this interval expires, ICON generates an error message and forces the transaction to be rolled back. Default Value: `600` Valid Values: Any integer Changes Take Effect: Immediately **Examples:** • `dbw-request-tout = 30` • `dbw-request-tout = 120`

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
Database writing (continued)	dbw-seq-step	Specifies the reservation value that ICON uses when updating the counter in the `SEQCOUNTER` field of the `G_PROV_CONTROL` table. At startup, ICON reads the initial counter value (*M*) from the `G_PROV_CONTROL` table, increments the counter in every database transaction, and writes the new value into the `GSYS_SEQ` or `GSYS_USEQ` field of the tables that are participating in the transaction. ICON updates the value of the `SEQCOUNTER` field in the `G_PROV_CONTROL` table as follows: 1. During the first database transaction after startup, ICON inserts the sum (*L*=*M*+*N*) of the initial counter value (*M*) and the reservation value specified by the `dbw-seq-step` option (*N*). For example, if the initial value that ICON retrieves at startup is `700`, and if you keep the default value of `500` for the `dbw-seq-step` option, ICON writes `1200` during the first transaction. 2. During the next *N*–*1* transactions, ICON does not update the `G_PROV_CONTROL` table, but updates only those tables that are participating in the transactions. 3. During the *N*th transaction, ICON inserts into the `G_PROV_CONTROL` table a new value (*K*=*L*+*N*) that is the sum of the current counter value (*L*) set in Step 1 and the reservation value (*N*). Continuing the example from Step 1, during the 500th transaction, ICON writes the new counter value of `1700`. 4. During each subsequent *N*th transaction, ICON uses the same logic to update the value of the `SEQCOUNTER` field in the `G_PROV_CONTROL` table. Default Value: `500` Valid Values: Any integer Changes Take Effect: After restart **Examples:** • `dbw-seq-step = 100` • `dbw-seq-step = 300`

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
Database writing (continued)	dbw-seq-tout	Specifies the amount of time, in seconds, after which ICON writes the current value of the transaction counter to the `G_PROV_CONTROL.SEQCURRENT` field. The merge procedure relies on this field for the detection of newly updated records. (For more information, see the section about the merge stored procedure in the *Interaction Concentrator 7.6 User's Guide.*) Default Value: `60` Valid Values: Any integer Changes Take Effect: Immediately **Examples:** • `dbw-seq-tout = 30` • `dbw-seq-tout = 120`
ICON Role	role	Specifies the type of data that this ICON instance processes and stores in IDB. Default Value: `all` Valid Values: A comma-separated list including any of the following: `svc` Stores service information about this ICON instance, for identification purposes, in IDB, The `svc` role is predefined and cannot be turned off regardless of other roles assigned to the ICON application. `cfg` Stores the initial configuration state and a history of configuration changes retrieved from Configuration Server. `gcc` Stores interaction-related and party-related information—that is, T-Server and Interaction Server data that pertains to voice and Multimedia interactions, and the parties associated with those interactions. `gls` Stores T-Server and Interaction Server data that pertains to agent states and agent login sessions. `gud` Stores T-Server and Interaction Server data that pertains to the attached data associated with calls. `gos` In an environment with the Outbound Contact solution, stores OCS data that pertains to outbound calls and campaigns. all Stores all types of data. Changes Take Effect: After restart

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
ICON Role	role (continued)	Any combination of the valid values can be used. Prefixing an option value with a tilde (~) excludes that type of data from ICON processing, and includes all other types (except for the `gcc` role for T-Server interactions only. See **Note 1** below). For example, the value `~cfg` deactivates ICON processing of configuration data, and activates processing and storage of all other types of data. Every ICON instance always stores its service information, regardless of whether you set the `svc` value. **Note 1:** If connected to T-Server, the ICON instance will perform the `gcc` role, regardless of whether you have configured it to perform this role or not. **Note 2:** An ICON instance with a configured connection to Interaction Server will not perform the `gcc` role if you set the option value to `~gcc`. Similarly, if the role option does not define any of the following roles—`gcc`, `gud`, `gls`— then ICON will not process interactions even if Interaction Server is included in the connection section of ICON application in Configuration Manager. **Note 3:** Ensure that the role that you specify for the ICON instance is consistent with the role that you specify for the DAP (see page 116). **Examples:** • `role = cfg, gcc, gud` • `role = all` • `role = gcc, gud, gls, gos` • `role = ~cfg`
IDB	db-schema-name	Specifies the database schema name ICON will use when the RDBMS requires an explicit schema name to be specified when executing stored procedures. For information about what the term *schema name* means and for any delimiters that the RDBMS syntax requires, see the vendor documentation for your RDBMS. Default value: Empty Valid values: Any string Changes Take Effect: After restart
In-memory queue	acc-proc-tout	Specifies the interval, in milliseconds, at which ICON scans its in-memory queue in order to determine whether the timeout set by the `acc-queue-lifespan` option has expired. Default Value: `3000` Valid Values: Any positive integer Changes Take Effect: Immediately

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
In-memory queue (continued)	acc-queue-lifespan	Specifies the interval, in seconds, at which ICON accumulates records in its in-memory queue before writing them to a persistent queue (as the first stage of serialization). The process of writing to a persistent queue is triggered when the limit set either by this option or by the `acc-queue-size` option is exceeded. Default Value: `5` Valid Values: Any positive integer Changes Take Effect: Immediately
	acc-queue-size	Specifies the maximum number of serialization records that ICON keeps in the in-memory queue before writing them to a persistent queue (as the first stage of serialization). The process of writing to a persistent queue is triggered when the limit set either by this option or by the `acc-queue-lifespan` option is exceeded. This option also defines the size of a database writing transaction. Default Value: `500` Valid Values: Any positive integer Changes Take Effect: Immediately
Multimedia	calls-in-the-past	Specifies whether ICON stores data for Multimedia interactions that begin while ICON is down, or while ICON has no connection to Interaction Server. The data stored for reconstructed interactions is the same as the data stored for the interactions that ICON tracks from their beginning. **Note:** ICON cannot restore a correct timestamp of interaction record creation, or the information about previous parties, or the first values of user data keys. Default Value: `false` Valid Values: `true` When this option is set to `true`, ICON reconstructs operational data about a Multimedia interaction that is already in progress when ICON receives one or more of the following reporting events from Interaction Server: • `EventPlacedInQueue` • `EventPlacedInWorkbin` • `EventAgentInvited` • `EventPartyAdded` `false` When this option is set to `false`, ICON does not record data for Multimedia interactions that begin while ICON is down, or while ICON has no connection to Interaction Server.

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
Multimedia	calls-in-the-past (continued)	Changes Take Effect: After restart
Open Media	mcr-om-processing	Specifies whether ICON stores information about Open Media interactions in IDB. By default, ICON processes interactions other than chat, e-mail, or voice and stores the type of media in special fields of the following tables: • GX_SESSION_ENDPOINT • G_AGENT_STATE_HISTORY • GS_AGENT_STAT • G_AGENT_STATE_RC • G_CALL With the 0 setting, ICON processes neither interactions nor agent data for Open Media. Default Value: 1 Valid Values: 0 ICON does not store data in IDB about interactions other than chat, e-mail, or voice. 1 ICON stores information in IDB about Open Media interactions. Changes Take Effect: After restart **Note:** For more information about Open Media support, refer to the chapter about integrating with Genesys Multimedia in the *Interaction Concentrator 7.6 User's Guide.*
Operational memory	om-check-filter-flag	Specifies whether or not ICON stores strategy activity according to the value of the om-activity-report configuration option that is defined on the script object (of type simple routing). If the value is set to 0, ICON stores all strategy activity regardless of the value of the om-activity-report. Default Value: 1 Valid Values: 0, 1 Changes Take Effect: After restart

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
Operational memory (continued)	om-max-in-memory	Specifies the maximum number of keep-in-memory interactions that were placed in queues or interaction workbins (in units of one thousand). Default Value: `100` Valid Values: `1-2,000` (in units of one thousand) Changes Take Effect: After restart Keep the default value unless you are advised otherwise by Genesys Technical Support. If you need to change the option's value, use the following formula to calculate an approximate value for this option: `Size of available operational memory (K)/((1,000 + size of user data (K))*2)` where: `size of user data` = average size of the interaction user data that is attached to the interaction in Interaction Server. **Note:** An incorrect value for this option can affect ICON performance, or cause ICON to stop processing interactions.
	om-memory-optimization	Specifies whether or not memory usage will be optimized. Default Value: `false` Valid Values: `true` ICON optimizes memory usage according to the values that are set for the following options: `om-max-in-memory` and `om-memory-clean`. `false` Preserves legacy behavior (prior to ICON release 7.6.1).

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
Operational memory	om-memory-optimization (continued)	Changes Take Effect: After restart
Outbound metrics	gos-write-duplicate-metrics	Specifies whether all metrics related to active outbound objects are stored in IDB exactly as Outbound Contact Server (OCS) provides them, or whether ICON filters out duplicate metrics. ICON identifies active outbound objects by `CampaignGUID`, `ChainGUID`, and `CallAttemptGUID`. Default Value: `0` Valid Values: `0` ICON does not subsequently write the same precalculated OCS metric after it is stored in IDB. `1` ICON writes all metrics related to active objects, exactly as OCS provides them, without filtering out possible duplicate metrics. Changes Take Effect: After restart **Note:** For more information about outbound-related metrics, refer to the chapter about integrating with Outbound Contact in the *Interaction Concentrator 7.6 User's Guide.*

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
Outbound metrics (continued)	gos-write-metrics	Specifies whether ICON writes any precalculated OCS metrics to IDB. Default Value: 1 Valid Values: 0 ICON does not store any precalculated metrics that OCS provides. 1 ICON stores precalculated metrics that OCS provides. Changes Take Effect: After restart
	gos-write-metrics-only	Specifies whether ICON excludes from database storage all outbound data except precalculated metrics. Default Value: 0 Valid Values: 0 ICON stores both OCS data and precalculated OCS metrics, regardless of the gos-write-metrics option value. 1 Provided that the gos-write-metrics option is also set to 1, ICON stores precalculated metrics, but not the other data that OCS provides. Changes Take Effect: After restart
Persistent queue	agent-pstorage-name	Specifies the name of the persistent cache file that ICON creates and uses to store information about agent login sessions before writing the information to IDB. Default Value: apstorage.db Valid Values: Any valid file name Changes Take Effect: After restart

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
Persistent queue (continued)	pq-backlog-alarm-threshold	Specifies the maximum number of records allowed to be pending in the persistent queue for submission to IDB. When the threshold is reached, ICON generates log message `25025`. The purpose of the option is to enable an alarm to be generated when the number of records not submitted to IDB is unacceptably high because of some failure in the environment. The following are examples of environment failure: • The database is not available, or it is not responding to ICON requests. • The load on the ICON server is too high. • The ICON process has not been suitably configured (for example, large quantities of expensive attached data are being stored). • The network is slow. • The load on the RDBMS is too high. • There is an overall system overload. To avoid triggering the alarm because of expected fluctuations in the ICON server load, do not set the value of this option too low. The optimal value depends on your specific deployment and contact center activity profile. Genesys recommends basing the value on the average load in your contact center, calculated from reported values for `Records queued in previous 15 minutes` on the `Database Writer` performance counter page (see the *Interaction Concentrator 7.6 User's Guide*). For example, if 100,000 records are queued during 15 minutes of average load, consider setting the `pq-backlog-alarm-threshold` value to `400,000`, to cover one hour of average load and allow for some peak loads. Default value: `0` Valid values: `0`–(2^{32}–`1`) (`0` indicates that no log message will be generated) Changes Take Effect: Immediately
	pq-backlog-clearance-threshold	Specifies the minimum number of records pending in the persistent queue at which ICON will generate message `25026`, if log message `25025` was previously generated (see `pq-backlog-alarm-threshold`). Default value: `0` Valid values: 0–value of `pq-backlog-alarm-threshold` (`0` indicates that no log message will be generated) Changes Take Effect: Immediately

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
Persistent queue (continued)	pq-dbname	Specifies the name of the persistent queue file that ICON creates and uses to store information before writing the information to IDB. With the default setting, the file name consists of the prefix `icon_`, followed by the identifier that Configuration Server assigns to this particular ICON application (DBID)—for example, `icon_161.pq`. The special value `:memory:` instructs the Persistent Queue Manager to use memory as storage instead of a physical file. Using memory for persistent queue storage may improve ICON performance with regard to database writes. However, this setting increases memory consumption, and you run the increased risk of losing data in the event ICON terminates abnormally. **Note:** Do not use the `:memory:` value if the `role` option for the ICON instance is `cfg`. By design, configuration synchronization requires persistent storage, so the temporary storage provided by `pd-dbname` = `:memory:` will generate configuration synchronization errors for an ICON configured to perform the `cfg` role. Default Value: `icon_<DBID>.pq` Valid Values: Any valid file name Changes Take Effect: After restart **Note:** Genesys recommends that this file reside locally—not on a network.
	pq-purge-number	Specifies the number of committed transactions after which ICON purges from its persistent queue the information that is already stored in IDB. For example, if the value is set to `10`, ICON performs a purge operation on its persistent queue after every ten transactions. Default Value: `10` Valid Values: Any positive integer Changes Take Effect: Immediately
	pq-startup-check	Specifies whether ICON checks the integrity of its persistent queue at startup. With a large-sized persistent queue file (hundreds of megabytes), the integrity check takes up to three minutes of startup time. For any integrity violations that it finds during the integrity check, ICON issues an error message, changes the extension of the corrupted queue file to `*.bak`, and creates a new database queue. Default Value: `true` Valid Values: `true, false` Changes Take Effect: After restart

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
Scenario recognition	gcti-mode-monitoring	Regulates the mode that ICON uses for multi-site scenario recognition. Default Value: `0` Valid Values: `1` Enables new scenario recognition logic that is implemented in ICON release 7.6.000.21. `0` Preserves ICON legacy behavior (prior to release 7.6.000.21). Changes Take Effect: After restart
Synchronization	start-cfg-sync	Specifies whether ICON performs synchronization of configuration data between Configuration Database and IDB. By default, ICON ignores this option. To start data synchronization, first set the option value to `0`; then, change the option value to `1`. This action prompts ICON to start the synchronization process. Once started, the synchronization process completes regardless of the subsequent changes to the option value. **Note:** To perform data synchronization, ICON must have a connection to Configuration Server from the moment you change the option value from `0` to `1` until the moment when data synchronization is complete. Default Value: `-1` Valid Values: `-1` ICON ignores this option even when it is defined in the configuration. `0` ICON acknowledges that this option is specified in the configuration and waits for a notification about the option value change from `0` to `1`. `1` ICON starts the data synchronization between Configuration Database and IDB under the condition that the value changed first to `0` and then from `0` to `1` during ICON run time. The value of `1` at ICON startup does not trigger the synchronization of configuration data. Changes Take Effect: Immediately upon real-time notification only

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
Synchronization (continued)	sync-call-data-limit	Specifies the maximum number of pending synchronizations for calls and attached data. This option controls memory consumption during the process of synchronizing calls and user data. The call record is not terminated in IDB until all attached data related to that call has been written to the database. Until then, ICON keeps in memory all information that is related to the call. If the limit is reached, no more call records will be locked until the number of pending synchronizations falls below the configured limit. This situation does not produce any loss or duplication of data, but call records that have not been locked might be marked as terminated before their related attached data has been written to IDB. **Note:** Genesys recommends that you do not change the default value. Default value: `1000000` Valid values: `0`–(2^{32}–1) (`0` indicates that no synchronization takes place) Changes Take Effect: Immediately
	tsync-threshold	Specifies the maximum time difference, in milliseconds, allowed between the ICON host and the T-Server (or, if applicable, Interaction Server) host. When the threshold is reached, ICON generates standard log message `25130`. See also the `min-tsync-roundtrip` option in the `Switch` object (page 110). Default Value: `1000` Valid Values: `0`–`2000` (`0` indicates that no log message will be generated) Changes Take Effect: Immediately

Table 4: ICON Configuration Options—callconcentrator Section (Continued)

Area of Functionality	Option	Description
Virtual Queue	vq-write-mode	Specifies how ICON writes to IDB information about a particular association between an interaction and a virtual queue. When this option is set to `0`, ICON creates a complete IDB record when the association is terminated, as indicated by either `EventDiverted` or `EventAbandoned`. When this option is set to `1`, ICON initially creates an IDB record when the association starts, as indicated by the `EventQueued` TEvent; after the association is terminated, as indicated by either `EventDiverted` or `EventAbandoned`, ICON updates the existing record. Default Value: `0` Valid Values: `0` ICON stores virtual queue–related data in one step. `1` ICON stores virtual queue–related data in two steps. Changes Take Effect: After restart
	extended-route-result	Specifies whether ICON stores extended routing results—statuses of interactions distributed by URS 7.6—in IDB. Default Value: `0` Valid Values: `0` ICON stores route results in `G_VIRTUAL_QUEUE` and `G_ROUTE_RESULT` IDB tables as implemented in ICON release 7.5. `1` ICON stores extended routing results in `G_VIRTUAL_QUEUE` and `G_ROUTE_RESULT` IDB tables as implemented in ICON release 7.6 (see Note 2). Changes Take Effect: After restart **Note 1:** For details of the routing results stored in IDB when `extended-route-result` = `0` or `1`, refer to the chapter about monitoring virtual queues and route points in the *Interaction Concentrator 7.6 User's Guide.* **Note 2:** The new feature implemented in Interaction Concentrator 7.6 related to writing extended routing results into IDB (`G_ROUTE_RESULT` and `G_VIRTUAL_QUEUE` IDB tables) requires: • Universal Routing Server (URS) release 7.6. • URS configuration options `report_reasons` and `report_targets` set to `true`.

custom-states Section

You must name this section `custom-states` in the configuration.

The options in this section configure Interaction Concentrator support for the processing of custom agent states and common user data.

Table 5 describes the options in the `custom-states` section. For ease of reference, the table presents the options in alphabetical order.

Table 5: ICON Configuration Options—custom-states Section

Option	Description
AgentRecordUserTypes	Defines the custom agent states. The agent desktop application starts and ends custom agent states, and it sends the required key-value pair (KVP) data to ICON through T-Server's `EventUserEvent`. ICON verifies the values provided in `EventUserEvent` for the key names specified by this configuration option, in order to determine when custom states start (value for the configured key = "`+`") and finish (value = "`-`"). After a state is started and before it is finished, the desktop application can send data in user events, to be stored in the custom fields that correspond to the state, as specified by the `AgentUserFields` configuration option. For more information about ICON custom state recording, see the section about using custom states in the *Interaction Concentrator 7.6 User's Guide.* Default Value: No default value Valid Values: A comma-separated list of the custom state codes and key names in the format `<StateCode>,<KeyName>`. The custom state code must be a number greater than `199`. Changes Take Effect: Immediately **Example:** `AgentRecordUserTypes = 207,AfterCallWork,208,Break`

Table 5: ICON Configuration Options—custom-states Section (Continued)

Option	Description
AgentUserFields	Specifies the fields in the G_CUSTOM_STATES table in which ICON will store values (provided in the UserData section of EventUserEvent) for the specified key names, for data that was sent while the DN was in a custom agent state. Default Value: No default value Valid Values: A comma-separated list of the data types, table field names, and key names in the format <Type>,<FieldName>,<KeyName>. **Note:** All the custom data fields in the G_CUSTOM_STATES table require character-type data. Regardless of the data type that you specify in this option, ICON converts the value from the UserData KVP into a string, before storing it in the custom data field that is specified for that key name. If the value of the key in the UserData KVP is KVList, ICON ignores the value. Changes Take Effect: After restart **Example:** AgentUserFields = char,CUST_DATA_1,KeyName1,char,CUST_DATA_2,KeyName2 • The value of the key with the name KeyName1 will be stored in the CUST_DATA_1 field. • The value of the key with the name KeyName2 will be stored in the CUST_DATA_2 field.
EventData	Specifies the list of key names for which ICON will store KVP data (provided in the UserData section of EventUserEvent) in the G_CUSTOM_DATA_S table. Default Value: No default value Valid Values: A comma-separated list of the data types and key names in the format <Type>,<KeyName>. **Note 1:** The limit for option specifications in Configuration Manager is 255 characters. If your desired EventData option specification exceeds this limit, you can specify additional options in the format EventData_X, where X is any integer, starting from 1. ICON recognizes all the EventData specifications as one option, and it concatenates the content of the options in sequence. **Note 2:** The field for the key's value in the G_CUSTOM_DATA_S table requires character-type data. Regardless of the data type that you specify in this option, ICON converts the value from the UserData KVP into a string, before storing it in the G_CUSTOM_DATA_S table. If the value of the key in the UserData KVP is KVList, ICON ignores the value. **Note 3:** Ensure that the key name you specify does not conflict with a key name specified in the GlobalData option (see page 90). The key names specified in the EventData and GlobalData options must be unique. Changes Take Effect: Immediately **Example:** EventData = char,CUSTOMER_NAME,int,CUSTOMER_PHONE

Table 5: ICON Configuration Options—custom-states Section (Continued)

Option	Description
GlobalData	Specifies the list of key names for which ICON will store KVP data (provided in the `UserData` section of `EventUserEvent`) in the `G_CUSTOM_DATA_P` table. The position of the key name in the list determines the mapping to the custom data field in the `G_CUSTOM_DATA_P` table. Default Value: No default value Valid Values: A comma-separated list of the data types and key names in the format `<Type>,<KeyName>`. **Note 1:** The limit for option specifications in Configuration Manager is 255 characters. If your desired `GlobalData` option specification exceeds this limit, you can specify additional options in the format `GlobalData_X`, where `X` is any integer, starting from `1`. ICON recognizes all the `GlobalData` specifications as one option, and it concatenates the content of the options in sequence. **Note 2:** All the custom data fields in the `G_CUSTOM_DATA_P` table require character-type data. Regardless of the data type that you specify in this option, ICON converts the value from the `UserData` KVP into a string, before storing it in the `G_CUSTOM_DATA_P` table. If the value of the key in the `UserData` KVP is `KVList`, ICON ignores the value. **Note 3:** Ensure that the key name you specify does not conflict with a key name specified in the `EventData` option (see page 89). The key names specified in the `EventData` and `GlobalData` options must be unique. Changes Take Effect: Immediately **Example:** `GlobalData = char,CUSTOMER_NAME,int,CUSTOMER_PHONE` • The value of the key with the name `CUSTOMER_NAME` will be stored in the `CUST_DATA_1` field. • The value of the key with the name `CUSTOMER_PHONE` will be stored in the `CUST_DATA_2` field.

Table 5: ICON Configuration Options—custom-states Section (Continued)

Option	Description
store-event-data	Specifies what, if any, KVP data (provided in the `UserData` section of `EventUserEvent`) ICON will store in the `G_CUSTOM_DATA_S` table. Default Value: `none` Valid Values: `none` ICON does not store any data. `all` ICON stores the values of all keys. `conf` ICON stores the values of the keys that are configured in the `EventData` option. Changes Take Effect: Immediately **Example:** `store-event-data = conf`

dbw-error-reactions Section

You must name this section `dbw-error-reactions` in the configuration.

The options in this section define Interaction Concentrator reactions to specific database error messages. In other words, each configuration option in this section represents a rule for handling a certain database error.

If Interaction Concentrator receives an database error message, it will try to find the text, specified in the configuration option as a substring of the error message. If it finds this substring text, the specified error reaction will be applied.

You can specify any number of options within this configuration section. For example, the `dbw-error-reactions` section that you define might appear as follows:

```
[dbw-error-reactions]
dbw-error1="error=ORA1123-005;reaction=retry"
```

Preconfigured Error

The Interaction Concentrator application template includes a preconfigured error reaction to ignore unique constraint violations:

Option Name: `uniqueness`

Option Value: `error=unique;reaction=ignore`

Table 6 describes the options in the `dbw-error-reactions` section.

Table 6: ICON Configuration Options—dbw-error-reactions Section

Option	Description
<error-name>	Defines how ICON reacts to a database error message that contains a particular text substring. Create a separate option for every database error message for which a certain reaction is required. Specify any meaningful name as the option name, making it unique within the `dbw-error-reactions` section; ICON does not process the name parameter. Include both a database error message and the expected reaction as two parameters of the option value, in the following format: `error=<error substring>;reaction=<reaction type>` Where: `<error substring>` — The database error message or a substring of the error message that is sufficient to identify it among all database error messages. ICON selects the first option with a matching substring among all options that contain matching substrings. You can use any character and symbol in `<error substring>` except the semicolon (`;`). A semicolon signals the end of the `error` parameter to ICON. If you must include a semicolon within a substring, surround the entire substring with single quotation marks (`'error substring'`) or double quotation marks (`"error substring"`). `<reaction type>` — The expected reaction to the database error message identified by `<error substring>`. The reaction can be one of the following: • `reconnect`—ICON forcefully disconnects from the database and attempts to reconnect after receiving the database error message identified by `<error substring>`. This reaction type is recommended for error messages related to the temporary unavailability of a database that is inadequately processed by the database server. • `retry`—ICON rolls back the current transaction, and then attempts to resubmit it after receiving the database error message identified by `<error substring>`. This reaction type is recommended for error messages related to nonfatal database problems (for example, a locked table state) that tend to disappear during subsequent transaction attempts. • `ignore`—ICON rolls back the current transaction after receiving the database error message identified by `<error substring>`. ICON then attempts to resubmit the failed transaction, statement by statement, ignoring the statement that caused the error. This reaction type is recommended for logical errors such as constraint violations.

Table 6: ICON Configuration Options—dbw-error-reactions Section (Continued)

Option	Description

<error-name> (continued)	Default Value: No default value
	Valid Values: Any string in the following format:
	`error=<error substring>;reaction=<reaction type>`
	Changes Take Effect: Immediately
	Example:
	The following configuration option prompts ICON to resubmit a request that previously failed with an error message that contains the substring `ORA1123-005`:
	`dbw-error1="error=ORA1123-005;reaction=retry"`

filter-data Section

You must name this section `filter-data` in the configuration.

The options in this section control Interaction Concentrator output to IDB. Review the options in this section carefully and refer to the *Interaction Concentrator 7.6 Physical Data Model* for your RDBMS type for details about data stored in the IDB tables that are mentioned in the option descriptions. Evaluate whether your reports require each type of described data. Excluding certain types of data from IDB storage may help you save database space, and thus improve your database performance.

Table 7 describes the options in the `filter-data` section. For ease of reference, the table presents the options in alphabetical order.

Table 7: ICON Configuration Options—filter-data Section

Option	Description
acd-party-history	Specifies whether ICON should exclude, from IDB storage, party history information about distribution devices—such as ACD queues, Routing Points, and virtual routing points. By default, ICON collects party history information about distribution devices and stores this information in the `G_PARTY_HISTORY` IDB table. With a setting of `1`, ICON does not store party history information in the `G_PARTY_HISTORY` table. **Note:** The `acd-party-history` option applies to SIP and voice interactions only. Default Value: `0` Valid Values: `1, 0` Changes Take Effect: After restart

Table 7: ICON Configuration Options—filter-data Section (Continued)

Option	Description
acd-party-metrics	Specifies whether ICON should exclude, from IDB storage, party metrics for distribution devices—such as ACD queues, Routing Points, and virtual routing points. By default, ICON collects pre-calculated party metrics for distribution devices and stores this information in the `G_PARTY_STAT` IDB table. With a setting of `1`, ICON does not store data in the `G_PARTY_STAT` table for distribution devices. **Note:** The `acd-party-metrics` option applies to SIP and voice interactions only. Default Value: `0` Valid Values: `1, 0` Changes Take Effect: After restart
call-history	Specifies whether ICON should exclude call-history information from IDB storage. By default, ICON collects and stores call history data in the `G_CALL_HISTORY` IDB table. With a setting of `1`, ICON ceases writing to this table. Default Value: `0` Valid Values: `1, 0` Changes Take Effect: After restart
call-metrics	Specifies whether ICON should exclude call metrics from IDB storage. By default, ICON calculates call metrics and stores them in the `G_CALL_STAT` IDB table. With a setting of `1`, ICON ceases writing to this table. Default Value: `0` Valid Values: `1, 0` Changes Take Effect: After restart
external-party	Specifies whether ICON should exclude external-party data from IDB storage. By default, ICON collects information about external parties (for example, interaction participants outside a given switch domain) and stores this information in the following IDB tables: • `G_PARTY` • `G_PARTY_HISTORY` • `G_PARTY_STAT` With a setting of `1`, ICON collects and stores data about internal parties only (for example, interaction participants within a given switch domain). Default Value: `0` Valid Values: `1, 0` Changes Take Effect: After restart

Table 7: ICON Configuration Options—filter-data Section (Continued)

Option	Description
gls-all	Specifies whether ICON should exclude all information about agent activity from IDB storage. By default, ICON collects information about agent activity— such as login sessions, agent state—unless certain types of data are configured to be excluded by setting one or more of the following options to `1`: • `gls-ivr` (see page 96) • `gls-no-person` (see page 97) • `gls-queue` (see page 97) • `gls-wm` (see page 98) ICON stores this information in the following IDB tables: • `G_LOGIN_SESSION` • `GX_SESSION_ENDPOINT` • `G_AGENT_STATE_HISTORY` • `G_AGENT_STATE_RC` • `G_DND_HISTORY` • `GS_AGENT_STAT` • `GS_AGENT_STAT_WM` With a setting of `1` for the `gls-all` option, ICON ceases writing to these tables. Note, however, that with a `1` setting, ICON continues writing agent's ID to the `G_PARTY` table. Default Value: `0` Valid Values: `1, 0` Changes Take Effect: After restart

Table 7: ICON Configuration Options—filter-data Section (Continued)

Option	Description
gls-ivr	Specifies whether ICON should exclude, from IDB storage, data about agent activity at IVR endpoints. By default, ICON collects data about agent activity when agent login sessions are initiated from IVR endpoints and stores this information in the following IDB tables: • G_LOGIN_SESSION • GX_SESSION_ENDPOINT • G_AGENT_STATE_HISTORY • G_AGENT_STATE_RC • G_DND_HISTORY • GS_AGENT_STAT • GS_AGENT_STAT_WM With a setting of 1, ICON verifies whether the DN at which an agent logs in is an IVR device; in this case, ICON does not store information about this agent's activity to these tables. Furthermore, for parties associated with an IVR device, ICON does not record the agent's ID in the G_PARTY IDB table. Default Value: 0 Valid Values: 1, 0 Changes Take Effect: After restart **Note:** See page 113 for more information about setting the ivr option to configure a DN as an IVR port. For a description of how ICON identifies an IVR, see the *Interaction Concentrator 7.6 User's Guide.*
gls-metrics	Specifies whether ICON should exclude agent states from IDB storage. By default, ICON collects agent states unless certain types of data are configured to be excluded by setting one or more of the following options to 1: • gls-all (see page 95) • gls-ivr (see page 96) • gls-no-person (see page 97) • gls-queue (see page 97) • gls-wm (see page 98) ICON stores agent states information in the following IDB tables: • GS_AGENT_STAT • GS_AGENT_STAT_WM With a setting of 1, ICON does not store information about agents states to these tables. Default Value: 0 Valid Values: 1, 0 Changes Take Effect: After restart

Table 7: ICON Configuration Options—filter-data Section (Continued)

Option	Description
gls-no-person	Specifies whether ICON should exclude, from IDB storage, data about agent activity for agents whose login ID is not associated with any `Person` configuration object. By default, ICON collects data about all agent activity and stores this information in the following IDB tables: • `G_LOGIN_SESSION` • `GX_SESSION_ENDPOINT` • `G_AGENT_STATE_HISTORY` • `G_AGENT_STATE_RC` • `G_DND_HISTORY` • `GS_AGENT_STAT` • `GS_AGENT_STAT_WM` With a setting of `1`, ICON verifies whether the `LoginID` reported in events regarding agent states is assigned to any `Person` object configured in the Configuration Database; if this is not the case, ICON does not store information about this agent's activity to these tables. Default Value: `0` Valid Values: `1`, `0` Changes Take Effect: After restart
gls-queue	Specifies whether ICON should filter out information, from IDB storage, about the queues where agents are logged in. By default, ICON collects information about agents' queue(s) and stores this information in the following IDB tables: • `G_AGENT_STATE_HISTORY` • `G_AGENT_STATE_RC` • `GS_AGENT_STAT` • `GS_AGENT_STAT_WM` • `GX_SESSION_ENDPOINT` With a setting of `1`, ICON ceases writing queue-related data to the first four tables (above). ICON will continue writing information to the `GX_SESSION_ENDPOINT` table about the queues where agents are logged in. Default Value: `0` Valid Values: `1`, `0` Changes Take Effect: After restart

Table 7: ICON Configuration Options—filter-data Section (Continued)

Option	Description
gls-wm	Specifies whether ICON should exclude, from IDB storage, data about changes in agent work mode that do not coincide with changes in agent state. By default, ICON collects and stores data about agents' work mode and changes in agents' work modes in the following IDB tables: • G_AGENT_STATE_HISTORY • G_AGENT_STATE_RC • GS_AGENT_STAT_WM With a setting of 1, ICON ignores information about work mode and work mode changes. It records a value of unknown in the (above) IDB tables. **Note:** This option does not affect ICON's ability to track after-call work. Default Value: 0 Valid Values: 1, 0 Changes Take Effect: After restart
ir-history	Specifies whether ICON should exclude data about the interaction record history from IDB storage. By default, ICON collects interaction record history and stores this information in the G_IR_HISTORY IDB table. With a setting of 1, ICON ceases writing data to this table. Default Value: 0 Valid Values: 1, 0 Changes Take Effect: After restart
observer-party	Specifies whether ICON should exclude, from IDB storage, data related to a service observer on a call. By default, ICON collects data about every party involved with the call and stores this information in the following IDB tables: • G_PARTY • G_PARTY_HISTORY • GS_PARTY_STAT With a setting of 1, ICON does not store data about the party with the role Observer to these tables. Default Value: 0 Valid Values: 1, 0 Changes Take Effect: After restart

Table 7: ICON Configuration Options—filter-data Section (Continued)

Option	Description
udata-history-terminated	Specifies whether ICON should exclude, from IDB storage, information about changes in `UserData` values for certain keys. When ICON is configured in a way that it should store an entire history of `UserData` values for certain keys, ICON collects data about every change in value for those keys and, at interaction termination, stores this information in the following IDB tables: • `G_USERDATA_HISTORY` • `G_SECURE_USERDATA_HISTORY` With a setting of `1`, ICON does not insert new records in these tables at call termination time. ICON does, however, continue to write information about the creation, addition, and removal of key-value pairs to these tables. Default Value: `0` Valid Values: `1, 0` Changes Take Effect: After restart

listeners Section

You must name this section `listeners` in the configuration.

This section references a separate configuration section that describes the HTTP listening port.

Table 8 describes the options in the `listeners` section.

Table 8: ICON Configuration Options—listeners Section

Option	Description

`<http-connection>`	Points to the configuration of an HTTP connection to ICON. The option name must replicate the name of another configuration section in the ICON `Application` object. Because ICON does not process this option value, you can use it to enter a short description for the connection; ICON will print this description to its log.
	Default Value: No default value
	Valid Values: Any string
	Changes Take Effect: Immediately
	Example:
	`[listeners]`
	`http-9090="ICON HTTP listener"`
	Where `http-9090` is the option that describes the parameters of an HTTP connection at a port that ICON opens for listening.
	Note: To enable access to the performance counters, configure an HTTP Listener option, and then configure a corresponding section with its options.

<http-connection> Section

You can choose any name for this section, provided that it matches the name that you specify for one of the options in the `listeners` section. Use the options described in Table 9 to set parameters for this connection.

Example:

```
[http-9090]
port=9090
transport=tcp
protocol=http
```

Table 9: ICON Configuration Options—<http-connection> Section

Option	Description
port	Specifies the number of the port that ICON opens for HTTP listening. Default Value: No default value Valid Values: Any integer from `1` to `65535` Changes Take Effect: After restart **Warning!** The value for the `port` option must not coincide with the `ICON Application` object's communication port that is opened for client connections.
protocol	Specifies the application-level protocol for the configured listener. Change the value to `http` to enable access to interfaces that are exposed through HTTP in ICON and that display performance counters. **Note:** The HTTP interface is not available by default. Default Value: `sip` Valid Values: `http, sip` Changes Take Effect: Immediately
transport	Specifies the transport layer protocol for the connection between ICON and its client. Default Value: `TCP` (Transmission Control Protocol) Valid Value: `TCP` Changes Take Effect: Immediately

log Section

In addition to the log options that are common to all Genesys Server applications, and that are described in the *Framework 7.6 Configuration Options Reference Manual,* Interaction Concentrator supports a number of unique log options that can help you troubleshoot various scenarios when you deploy ICON and test its functionality in your environment.

Use the `log` section on the `Options` tab to set all of Interaction Concentrator's log options.

For the log options described in Table 10, the meaning of the valid values is as follows:

0	No troubleshooting-related logging
1	Logging of errors only
2	Detailed troubleshooting-related logging
3	Full details in troubleshooting-related logging

Table 10 describes the options in the log section. For ease of reference, the table presents the options in alphabetical order.

Table 10: ICON Configuration Options—log Section

Option	Description
x-conn-debug-open	Specifies the verbosity with which ICON logs messages that are related to network connections and disconnections at a transport protocol level. The value 0 disables troubleshooting-related logging, and the value 3 produces the most detailed logs. Any value that you set for this option supersedes the value set for the x-server-trace-level option, with regard to network connection messages. Default Value: As specified by the x-server-trace-level option Valid Values: 0, 1, 2, 3 Changes Take Effect: Immediately
x-conn-debug-select	Specifies the verbosity with which ICON logs messages that are related to incoming information at a transport protocol level. This option may significantly increase log volume. The value 0 disables troubleshooting-related logging, and the value 3 produces the most detailed logs. Any value that you set for this option supersedes the value set for the x-server-trace-level option, with regard to incoming information messages. Default Value: As specified by the x-server-trace-level option Valid Values: 0, 1, 2, 3 Changes Take Effect: Immediately
x-conn-debug-timers	Specifies the verbosity with which ICON logs messages that are related to triggering connection timers at a transport protocol level. The value 0 disables troubleshooting-related logging, and the value 3 produces the most detailed logs. Any value that you set for this option supersedes the value set for the x-server-trace-level option, with regard to connection timer–triggering messages. Default Value: As specified by the x-server-trace-level option Valid Values: 0, 1, 2, 3 Changes Take Effect: Immediately

Table 10: ICON Configuration Options—log Section (Continued)

Option	Description
x-conn-debug-write	Specifies the verbosity with which ICON logs messages that are related to outgoing information at a transport protocol level. The value `0` disables troubleshooting-related logging, and the value `3` produces the most detailed logs. Any value that you set for this option supersedes the value set for the `x-server-trace-level` option with regard to outgoing information messages. Default Value: As specified by the `x-server-trace-level` option Valid Values: `0, 1, 2, 3` Changes Take Effect: Immediately
x-server-config-trace-level	Specifies the verbosity with which ICON logs messages that are related to the configurations of the objects on which it relies. Messages can include configuration information about ICON's own `Application` object. The value `0` disables troubleshooting-related logging, and the value `3` produces the most detailed logs. Any value that you set for this option supersedes the value set for the `x-server-trace-level` option, with regard to configuration information messages. Default Value: As specified by the `x-server-trace-level` option Valid Values: `0, 1, 2, 3` Changes Take Effect: Immediately
x-server-dbw-trace-level	Specifies the verbosity with which ICON logs messages that are related to data-writing operations with the persistent queue and IDB. The value `0` disables troubleshooting-related logging, and the value `3` produces the most detailed logs. Any value that you set for this option supersedes the value set for the `x-server-trace-level` option, with regard to data-writing operation messages. Default Value: As specified by the `x-server-trace-level` option Valid Values: `0, 1, 2, 3` Changes Take Effect: Immediately
x-server-debug-level	Like `x-server-trace-level`, specifies the verbosity with which ICON prints troubleshooting-related logs. ICON supports both option names, but Genesys recommends using the `x-server-trace-level` option name. For more information, see the description of the `x-server-trace-level` option.

Table 10: ICON Configuration Options—log Section (Continued)

Option	Description
x-server-gcti-trace-level	Specifies the verbosity with which ICON logs messages that are related to its CTI communications. Messages can include `TEvents` that ICON receives from T-Server, including call-related and party-related events, and they can also include reports about CTI transactions. The value `0` disables troubleshooting-related logging, and the value `3` produces the most detailed logs. Any value that you set for this option supersedes the value set for the `x-server-trace-level` option, with regard to CTI communications messages. Default Value: As specified by the `x-server-trace-level` option Valid Values: `0, 1, 2, 3` Changes Take Effect: Immediately
x-server-trace-level	Specifies the verbosity with which ICON prints troubleshooting-related logs. This option sets the default value for all troubleshooting-related log options that are unique to ICON. That is, the value that you set for this option applies to the following function-specific options if you do not configure them: • x-conn-debug-open • x-conn-debug-select • x-conn-debug-timers • x-conn-debug-write • x-server-config-trace-level • x-server-dbw-trace-level • x-server-gcti-trace-level If you do set a value for any of these function-specific options, and if that value differs from the `x-server-trace-level` option value, the function-specific option value supersedes the `x-server-trace-level` option value for log messages that are related to that particular function. Default Value: `0` Valid Values: `0, 1, 2, 3` Changes Take Effect: Immediately

Switch Configuration Options

This section describes the configuration options that you configure on the `Annex` tab of the `Switch` configuration object that is related to your Interaction Concentrator. Interaction Concentrator processes these options.

gts Section

You must name this section `gts` in the configuration.

Table 11 describes the options in the `gts` section. For ease of reference, the table presents the options in alphabetical order.

Table 11: Switch Configuration Options—gts Section

Option	Description
call-deletion-timeout	Specifies the amount of time, in seconds, that ICON delays call context deletion after receiving a notification that the call has been deleted in T-Server. Default Value: `30` Valid Values: `3`–`600` Changes Take Effect: Immediately
delivered-flag	Controls when an unmonitored party is reconstructed (regarding an event flow), and when a transition to the `alerting` state occurs for this party in the call to an external destination (regarding the switch). Default Values: `0` (For all switches except Cisco CallManager) `1` (For Cisco CallManager) Valid Values: `0` The `alerting` state is generated when `EventDialing` arrives. `1` The `alerting` state is generated when `EventNetworkReached` arrives. `2` The `alerting` state is generated when `EventEstablished` arrives. `3` An unmonitored party is not reconstructed. **Note:** Genesys Technical Support recommends that you set the value of this option to `3` only for a particular event flow. Changes Take Effect: Immediately
emulate-event-queued-extrp	Enables the emulation of `EventQueued` for an External Routing Point that belongs to this switch. **Note:** Generation of `EventQueued` for an External Routing Point depends on a particular T-Server and its switch. ICON requires this event for correct party representation in any environment. Default Value: `1` Valid Values: `0` `EventQueued` is not emulated. `1` `EventQueued` is emulated. Changes Take Effect: Immediately **Note:** For help setting this option correctly, contact Genesys Technical Support.

Table 11: Switch Configuration Options—gts Section (Continued)

Option	Description
emulate-event-queued-rp	Enables the emulation of `EventQueued` for a Routing Point that belongs to this switch. **Note:** Generation of `EventQueued` for a Routing Point depends on a particular T-Server and its switch. ICON requires this event for correct party representation in any environment. Default Value: `1` Valid Values: `0` `EventQueued` is not emulated. `1` `EventQueued` is emulated. Changes Take Effect: Immediately **Note:** For help setting this option correctly, contact Genesys Technical Support.
emulate-event-queued-rq	Enables the emulation of `EventQueued` for a routing queue that belongs to this switch. **Note:** Generation of `EventQueued` for a routing queue depends on a particular T-Server and its switch. ICON requires this event for a correct party representation in any environment. Default Value: `1` Valid Values: `0` `EventQueued` is not emulated. `1` `EventQueued` is emulated. Changes Take Effect: Immediately **Note:** For help setting this option correctly, contact Genesys Technical Support.
fix-time-stamps	Enables adjustment of timestamps when the CTI event contains an earlier timestamp than the timestamp from a previously received CTI event. Default Value: `0` Valid Values: `0` Adjustment is disabled. Any non-zero integer Adjustment is enabled. Changes Take Effect: After restart

Table 11: Switch Configuration Options—gts Section (Continued)

Option	Description
gls-acw-first	Specifies, for this switch, which interaction ICON associates with after-call work (ACW). By default, ICON uses the setting of the `gls-acw-first` configuration option (described on page 69) as specified at the ICON application level. Setting this option to `1` enables ICON to associate after-call work with the voice interaction that immediately precedes the *start* of the after-call work (the first voice interaction). In this case, subsequent voice interactions that occur during the period of after-call work are considered as related to ACW processing and should not interrupt measurement of ACW-related metrics. When the agent logs out, changes his or her state to `Ready`, or goes `NotReady` for any reason other than to perform after-call work, ICON reports the end of the current ACW state. This option overrides an explicit setting of the `gls-acw-first` configuration option (described on page 69) at the ICON application level. **Note:** To associate the first ACW value, specify the value of this option on the switch `Application`. A change to the setting of this option on the ICON `Application` does not propagate to SIP switches. Default Value: `-1` Valid Values: `-1` ICON uses the value of the `gls-acw-first` option specified in the ICON `Application` object. If no value is set at the application level, ICON associates the last voice interaction with after-call work. `0` ICON associates the last voice interaction with after-call work. `1` ICON associates the first voice interaction with after-call work. Changes take effect: After restart

Table 11: Switch Configuration Options—gts Section (Continued)

Option	Description
gls-associations-rule	Controls, for this switch, how ICON associates DNs with a given agent login session. You can configure DN associations in Configuration Layer in two ways: • By adding DNs to the same `Place` object. (For example, a DN of `Position` type and DN of `Extension` type on the same phone set on Avaya switch must belong to the same `Place`. Another example involves DNs of different media types that are included into the same `Place`.) • By creating a relationship between two DNs through the `Association` field in the `DN Properties` window. The `gls-associations-rule` option enables ICON to process signaling on the associated DNs as follows: • With the setting of `-1`, ICON creates two separate login sessions for an agent who logs in with two different login IDs at two DNs that belong to the same place. For example, when one DN is used for multimedia interactions and another DN is used for voice interactions, ICON handles agent login sessions at these two DNs separately. • With the setting of `0`, ICON creates a single login session for two DNs that belong to the same place when an agent logs in at one of these DNs. For example, when an agent logs in at a position DN and an extension DN exists on the same phone set, ICON maintains a single login session for these two DNs. • With the setting of `1`, ICON creates a single login session for two DNs that are related through the `Association` field when an agent logs in at one of these DNs. For example, when an agent logs in to different queues from two associated DNs, ICON maintains a single login session for these two DNs. Default Values: `-1` (For SIP switches) `0` (For all switches except SIP) Valid Values: `-1` ICON associates each DN with a separate login session. `0` ICON associates a single login session with multiple DNs at a place. `1` ICON associates a single login session with two DNs associated through configuration. Changes take effect: After restart

Table 11: Switch Configuration Options—gts Section (Continued)

Option	Description
gls-enable-acw-busy	Specifies, for this switch, whether ICON should continue `ACW` and `NotReady` agent states when agents place or receive calls during the period of time that after-call work or `NotReady` agent state were invoked. The following IDB tables are affected by this option: `G_AGENT_STATE_HISTORY, G_AGENT_STATE_RC, GS_AGENT_STAT, GS_AGENT_STAT_VM`. For a description of these tables, refer to the IDB schema chapter in the *Interaction Concentrator 7.6 User's Guide.* Default Value: `1` Valid Values: `0` ICON continues `ACW` and `NotReady` agent states while an agent is handling another call. `1` ICON interrupts `ACW` and `NotReady` agent states while the agent handles another call. ICON recognizes completion of after-call work when any of the following occur: • The agent logs out. • The agent places himself/herself in `Ready` mode. • The agent goes `NotReady` for any reason other than to perform after-call work. (This includes indirect work mode changes such as when the agent walks away from his or her desk for a period of time.) Changes Take Effect: After restart **Note:** This option is not valid for SIP-compliant switches that handle interactions other than voice interactions.

Table 11: Switch Configuration Options—gts Section (Continued)

Option	Description
gls-flag-on-disconnect	Specifies how ICON handles agent states when disconnecting from, and reconnecting to, T-Server. Default Value: `0` Valid Values: `0` When reconnecting to T-Server, ICON compares the agent state from its memory with the state from `EventRegistered`. If the in-memory state does not match the currently reported agent state, ICON updates the agent state in both its internal memory and IDB. When disconnecting from T-Server, ICON performs no actions specific to agent states. `1` When disconnecting from T-Server, ICON closes any existing agent login sessions, and records this fact in IDB. When reconnecting to T-Server, ICON uses information from `EventRegistered` to start new agent login sessions, sets the current agent states, and writes this data to IDB. `2` When disconnecting from T-Server, ICON does not close any existing agent login sessions. Instead, it changes agents' states to `UNKNOWN`, and records these new states in IDB. When reconnecting to T-Server, ICON uses information from `EventRegistered` to restore the current agents' states and write them to IDB. Changes Take Effect: Immediately **Note 1:** Genesys recommends setting this option to `0` when the switch is monitored by T-Server 7.6. **Note 2:** Genesys recommends that you do *not* set the value of this option to `1` for deployments supporting HA of agent data. If you choose to set this option to `1`, however, a limited amount of HA agent data will be available (event sequence numbers only) provided that you also set the `gls-use-ts-id` configuration option (section [`gts`]) to `0` on the switch `Annex` tab.
gls-max-duration	Specifies the maximum amount of time, in hours, that an agent login session can last on a DN that belongs to this switch. Setting the option value to `0` (zero) prevents ICON from checking session durations. Default Value: `0` Valid Values: Any integer from `0` to `720` Changes Take Effect: Immediately

Table 11: Switch Configuration Options—gts Section (Continued)

Option	Description
gls-max-inactivity	Specifies the maximum allowed inactivity period, in hours, during a single login session. ICON closes any agent login session for which no agent-related activity is detected during the specified interval. Setting the option value to `0` (zero) prevents ICON from checking inactivity durations. Default Value: `0` Valid Values: Any integer from `0` to `72` Changes Take Effect: Immediately
gls-use-ts-id	Specifies whether ICON uses the login session ID generated by T-Server (GUID) or by itself when connecting to, or disconnecting from, T-Server. Default Value: `1` Valid Values: `0` ICON generates the login session ID itself. `1` ICON uses the login session ID (GUID) generated by T-Server. **Note:** If you set this option to `0`, make sure you also set the `gls-flag-on-disconnect` option to `1` in order to access available HA agent data. Changes Take Effect: After restart
min-tsync-roundtrip	Specifies the amount of time, in milliseconds, allowed for messages sent from ICON to T-Server to be acknowledged by T-Server, for the purposes of time synchronization. All messages that are acknowledged within the specified round-trip delay are considered valid for the purposes of calculating the time difference between the ICON host and the T-Server host. See also the `tsync-threshold` option in the ICON `Application` object (page 86). Default Value: `50` Valid Values: `0`–`500` (`0` indicates that no calculation will be performed) Changes Take Effect: Immediately

Table 11: Switch Configuration Options—gts Section (Continued)

Option	Description
sst-options	Specifies the TEvents that ICON uses to recognize a single-step transfer, in order to ensure the correct processing of scenarios involving a single-step transfer. Default Value: `0` Valid Values: `0` `EventReleased`, followed by a corresponding `EventRinging` or `EventQueued`. Arrivals of `EventReleased`, `EventRinging`, or `EventQueued` trigger the recognition logic. `1` `EventReleased` only. Arrival of `EventReleased` with an additional cause attribute triggers the recognition logic. **Note:** For T-Server for Siemens HiPath 4000 CSTA III, set the option value to `1`. Changes Take Effect: Immediately
support-dn-type-5	Enables the processing of events that pertain to DNs of the `Virtual Queue` type that belong to this switch. Default Value: `1` Valid Values: `0` ICON does not process any Virtual Queue–related events for DNs that belong to this switch. `1` ICON processes Virtual Queue–related events for DNs that belong to this switch. Changes Take Effect: Immediately

Table 11: Switch Configuration Options—gts Section (Continued)

Option	Description
suppress-user-data	Specifies whether the switch instructs T-Server to propagate attached data only when the attached data changes. This optimizes ICON processing of attached data by reducing network traffic. **Note:** This option can be set at the level of the switch or the ICON application (see `suppress-user-data` on page 73. ICON automatically detects the switch-level option setting. If the switch-level option is set to the (default) value of `1` (unchanged attached data suppressed), T-Server TEvents are optimized for all ICON applications that connect to the T-Servers for that switch. In this case, the switch-level option setting overrides any application-level settings of `0` (unchanged attached data not suppressed). If the switch-level option is set to `0`, an application-level setting of `1` will override it. Default Value: `1` Valid Values: `0` Unchanged attached user data is not suppressed. `1` Unchanged attached user data is suppressed. Changes Take Effect: After restart

DN Configuration Options

This section describes the ICON-related configuration options that you configure on the `Annex` tab of a `DN` configuration object. Interaction Concentrator processes these options.

gts Section

You must name this section `gts` in the configuration.

Table 12 describes the options in the `gts` section. For ease of reference, the table presents the options in alphabetical order.

Table 12: DN Configuration Options—gts Section

Option	Description
emulate-event-queued	Enables the emulation of `EventQueued` for this particular DN. This setting supersedes the value set in an `EventQueued`-related option at the `Switch` level. **Note:** Generation of `EventQueued` for a Routing Point, a Routing Queue, and an External Routing Point depends on a particular T-Server and its switch. ICON requires this event for correct party representation in any environment. Default Values: `0` For a Routing Queue DN `1` For a Routing Point DN and an External Routing Point DN Valid Values: `0` `EventQueued` is not emulated. `1` `EventQueued` is emulated. Changes Take Effect: Immediately
ivr	Specifies whether ICON treats this DN as an IVR port. By default, ICON identifies DNs as IVR ports using one of the following criteria: • DN has a type of `Voice Treatment Port` in Configuration Database. • DN has a type of `ACD Position` or `Extension`, and it is specified as an `Associated DN` in the properties of the IVR port at ICON startup time. **Note:** See the `gls-ivr` option description on page 96 to exclude data about agent activity associated with this IVR device from IDB storage. Default Value: `0` Valid Values: `0` ICON does not recognize this DN as an IVR port unless the DN configuration satisfies one of the above criteria. `1` ICON treats this DN as an IVR port regardless of other configuration parameters specified for this DN. Changes Take Effect: After ICON restart

Table 12: DN Configuration Options—gts Section (Continued)

Option	Description
monitor	Applicable to DNs of the `Virtual Queue` type, this option enables the processing of Virtual Queue–related events for this particular DN. This option is meaningful only when the `support-dn-type-5` configuration option is set to `1` (default) in the corresponding `Switch` object configuration (see page 111). Default Value: `1` Valid Values: `0` ICON does not handle any Virtual Queue–related events for this DN. `1` ICON processes Virtual Queue–related events for this DN. Changes Take Effect: Immediately

Script Configuration Options

This section describes the ICON-related configuration options that you configure on the `Annex` tab of `script` configuration objects of type `interaction queue` and `simple routing` (for a routing strategy). Interaction Concentrator processes these options.

callconcentrator Section

You must name this section `callconcentrator` in the configuration.

Table 13 describes the options in the `callconcentrator` section of `script` objects of type `interaction queue` and `simple routing`.

Table 13: Script Configuration Options

Option	Description
objects of type interaction queue	
om-memory-clean	Specifies whether or not ICON immediately removes an interaction from memory when the interaction is placed in the interaction queue. **Note:** The `om-memory-optimization` option must be set to `true` in order for `om-memory-clean` to work. Default Value: `0` Valid Values: `0` ICON does not take any special action to remove an interaction when it is placed in the interaction queue.

Table 13: Script Configuration Options (Continued)

Option	Description
om-memory-clean (continued)	1 ICON immediately removes an interaction when it is placed in the interaction queue. **Note:** Regardless of the value of this option, the removal of an interaction from ICON memory can be triggered by the value set for the `om-max-in-memory` option. Changes Take Effect: After restart
objects of type simple routing	
om-activity-report	Specifies whether or not ICON will store activity data that is related to a particular strategy. If the value is set to `false`, ICON will not store in IDB any data about parties for this strategy or any user data changes that are made by this strategy. **Note:** Regardless of the value of this option, ICON will always process `EventInteractionStopped` events. Default Value: `true` Valid Values: `true, false` Changes Take Effect: After restart

DAP Configuration Option

Table 14 describes the configuration option that you configure on the `Options` tab of a Database Access Point (DAP) `Application` configuration object that specifies the IDB connection. Interaction Concentrator processes this option.

Table 14: DAP Configuration Option

Option	Description
role	Specifies the type of data that ICON is allowed to store in IDB through this DAP. Use this option when you are writing different types of ICON data to different databases. Default Value: all Valid Values: A comma-separated list including any of the following: cfg — Stores the initial configuration state, and a history of configuration changes retrieved from Configuration Server. gcc — Stores call-related and party-related information—that is, T-Server data that pertains to calls and the parties associated with those calls. gls — Stores T-Server data that pertains to the agent states and agent login sessions. gud — Stores T-Server data that pertains to the attached data associated with calls. gos — In an environment with the Outbound Contact solution, stores OCS data that pertains to outbound calls and campaigns. Any combination of the valid values can be used. Prefixing an option value with a tilde (~) excludes that type of data from database storage through this DAP, and includes all other types. For example, the value ~cfg deactivates the storage of configuration data through this DAP, and activates processing and storage of all other types of data. All types of ICON data go through the same DAP in the following cases: • No role option is specified for the DAP. • The role option is specified, and its value is explicitly set to all. • You specified only one DAP Application object on the Connections tab of the ICON Application object. **Note 1:** Regardless of whether a given DAP handles all types of ICON data or a subset of them, a separate database connection is opened for each type of data. **Note 2:** Ensure that the role that you specify for the DAP is consistent with the role that you specify for the ICON instance (see page 76). Changes Take Effect: After restart **Examples:** • role = cfg,gcc,gud • role = all • role = ~cfg

Chapter

6 Troubleshooting ICON Installation and Deployments

This chapter describes problems that you might encounter when starting or running your Interaction Concentrator (ICON) application, and how to resolve them. It contains the following sections:

Startup Problems

The following are the most common startup problems:

- ICON does not connect to the Configuration Server (see "No Connection to the Configuration Server").
- ICON closes at startup (see "ICON Exits at Startup" on page 118).

No Connection to the Configuration Server

Possible causes of this problem are as follows:

- Command-line parameters on the ICON `Application` object's `Server Info` tab incorrectly specify the Configuration Server host and port.

  Solution: Correct the command-line parameters and restart the application. For more information about the command-line parameters, see "Command-Line Parameters" on page 58.

- Configuration Server is not running, or it is inaccessible over the network.

  Solution: Start Configuration Server or re-establish the network connection.

ICON Exits at Startup

See the ICON log file for the reasons for the startup failure. Possible reasons include:

- The application name specified in the ICON startup command line does not correspond to any existing `Application` object in the Configuration Layer.

  Solution: Create the `Application` object. For more information about creating and configuring the ICON `Application` see “Deploying Interaction Concentrator” on page 42.

- The application name specified in the ICON startup command line refers to an `Application` object that is not of the `Call Concentrator` application type.

  Solution: Remove the `Application` object of the incorrect type, and then use the correct template to create a new `Application` object of the `Call Concentrator` type. For more information about creating and configuring the ICON `Application` object, see “Deploying Interaction Concentrator” on page 42.

- There is no assignment to a Database Access Point (DAP) `Application` object on the `Connections` tab of the ICON `Application` object.

  Solution: Add to the ICON `Application` object’s `Connections` tab any DAP `Application` objects through which this ICON instance will access Interaction Databases (IDBs).

- The DAP `Application` object assigned on the ICON `Application` object’s `Connections` tab does not have an associated DB Server application.

  Solution: Associate a DB Server with the DAP `Application` object. For more information, see “Configuring a DAP” on page 53.

- The ICON instance has been configured to process call attached data (`role` = `gud`), but ICON cannot open the file specified in the `adata-spec-name` configuration option. The following error message in the log file indicates the existence of this condition:

  ```
  Std 02016 Unable to open attached data file '<attached data
  specification file name>', error code XXX
  ```

  Solution: Verify the following and correct as required.

  - The file specified in the `adata-spec-name` configuration option exists. If the file does not exist, create a new one or use the default attached data specification file (`ccon_adata_spec.xml`) provided in the Interaction Concentrator installation package.

  - The Interaction Concentrator user (the account under which ICON has been started) has the required permissions to read the attached data specification file.
- The persistent queue file has become corrupted.

  Solution: Force ICON to create a new persistent queue file by doing one of the following:
  - Using operating system commands, move or rename the corrupted `.pq` file. On restart, ICON will create a new `.pq` file with the original file name in the original location.
  - Reset the `pq-dbname` configuration option in the ICON `Application` object. On restart, ICON will create a new `.pq` file with the new file name in the specified location. For more information about the `pq-dbname` configuration option, see "pq-dbname" on page 84.

  In both cases, all unprocessed data in the old `.pq` file will be lost to ICON and IDB.
- There is no free disk space on the disk where the `apstorage.db` file resides.

  Solution: Free up memory on the disk or add more disk memory. For more information about the `apstorage.db` file, see the chapter about implementing high availability in the *Interaction Concentrator 7.6 User's Guide.*

Runtime Problems

The following are the most common runtime problems:

- ICON does not connect to T-Server or Interaction Server (see "No Connection to T-Server or Interaction Server" on page 120).
- ICON does not receive call-related events from T-Server (see "ICON Does Not Receive Call-Related Events from T-Server" on page 120).
- ICON does not write information to the database (see "ICON Does Not Write Information to the Database" on page 121).
- ICON has lost synchronization with the Configuration Database (see "ICON Has Lost Synchronization with the Configuration Database" on page 122).

No Connection to T-Server or Interaction Server

Possible causes of this problem are as follows:

- There is no assignment to the T-Server `Application` object or the Interaction Server `Application` object on the ICON `Application` object's `Connections` tab.

  Solution: Add to the ICON `Application` object's `Connections` tab any T-Server or Interaction Server `Application` objects from which this ICON instance will receive interaction-related information.

- The T-Server or Interaction Server application is not running, or it is not accessible over the network.

  Solution: Start the application or re-establish the network connection.

- The T-Server or Interaction Server `Application` object cannot connect to its `Switch` link.

  Solution: See the applicable troubleshooting guide for your particular T-Server or Multimedia Interaction Server.

- The release of the T-Server or Interaction Server `Application` object is not compatible with Interaction Concentrator. T-Server release 7.2 is the minimum version required by any release of Interaction Concentrator. Multimedia Interaction Server release 7.5 is the minimum version required for Interaction Concentrator support of Multimedia. For more information about Interaction Concentrator compatibility and interoperability with other Genesys components, see the first chapter in the Interaction Concentrator section of the *Genesys 7 Migration Guide*.

  Solution: Upgrade the T-Server or Interaction Server `Application` object to a compatible release.

- The `Switch` object associated with the T-Server `Application` object does not have all the necessary `DN` objects configured.

  Solution: Create the `DN` objects. For more information, see the *Deployment Guide* for your particular T-Server.

ICON Does Not Receive Call-Related Events from T-Server

Possible causes of this problem are as follows:

- ICON was not restarted after changes were made on the ICON `Application` object's `Connections` tab.

  Solution: Stop ICON, then restart. For more information, see "Starting and Stopping Interaction Concentrator" on page 57.

- There is no connection between the ICON `Application` object and T-Server. See "No Connection to T-Server or Interaction Server".

ICON Does Not Write Information to the Database

Possible causes of this problem are as follows:

- The database parameters are incorrectly specified on the DAP `Application` object. These parameters include the user name and password.

  Solution: Specify the correct values on the DAP `Application` object's `DB Info` tab, then restart ICON. For more information, see "Configuring a DAP" on page 53.

- DB Server is not running, or it is inaccessible over the network.

  Solution: Start DB Server or re-establish the network connection.

- The RDBMS server is not available, or the IDB to which DB Server is trying to connect is not available.

  Solution: Take the necessary steps to make the database server and database available.

- The DAP `Application` object has been configured for a role that prevents it from writing certain classes of information to the database.

  Solution: Reconfigure the `role` option for the DAP `Application` object. Restart ICON. For more information about configuring a DAP see "Configuring a DAP" on page 53, and a description of the `role` option on page 116.

- IDB has not been initialized by the Interaction Concentrator initialization scripts.

  Solution: Run the Interaction Concentrator initialization scripts. For more information, see "Creating IDB" on page 49.

- ICON was not restarted after changes were made on the ICON `Application` object's `Connections` tab.

  Solution: Stop ICON, then restart. For more information, see "Starting and Stopping Interaction Concentrator" on page 57.

- Records are accumulating in the in-memory queue and are not being written to IDB.

  Solution: This might not be a problem. Configuration options control whether a size threshold or timeout triggers the transfer of records from the in-memory queue to the persistent queue, from which the records are then written to IDB. Wait for the event that triggers the transfer, and re-evaluate your configuration as necessary. For more information about `In-memory queue` configuration options, refer to the option descriptions starting on page 77.

- The program logic consistently produces an error because of incorrect RDBMS settings. For example, there may be insufficient free space available on the RDBMS for data storage, or the rollback segment may be too small.

Solution: Review the error messages reported in the ICON log file. If you have configured an HTTP Listener, you can also view the error messages on the Database Writer performance counter web page (for more information, refer to the chapter about monitoring ICON performance in the *Interaction Concentrator 7.6 User's Guide*). Provide the appropriate fix on the RDBMS side. For example, if the error messages cite no free space available for data storage, increase the table space.

If the error was entirely related to the RDBMS problem, you do not need to restart ICON or perform any manipulation of the persistent queue (`.pq` file). However, if the `.pq` file has become corrupted and there are additional errors in the program logic, you must replace the `.pq` file (see the problem about a corrupted persistent queue file on page 119).

ICON Has Lost Synchronization with the Configuration Database

There are a number of reasons why ICON might lose synchronization with the Configuration Database, especially following a shutdown of ICON.

Loss of synchronization has the following impact on IDB:

- ICON fails to capture data about configuration objects created while ICON was stopped.
- ICON does not mark configuration data as deleted in cases where the applicable configuration objects were deleted while ICON was stopped.
- ICON fails to capture changes in associations between objects (while it is stopped).

Solution: If you suspect that your configuration data in IDB is inconsistent with Configuration Database, perform a manual resynchronization. See the chapter about resynchronizing configuration data in the *Interaction Concentrator 7.6 User's Guide.*

Merge Procedure Problems

The most common problems encountered in executing the merge procedure (`gsysIRMerge` or `gsysIRMerge2`) are as follows:

- The procedure fails to complete (see "Merge Procedure Does Not Complete Successfully").
- The procedure does not execute (see "Merge Procedure Does Not Execute" on page 124).
- Merge procedure performance is slow or constantly aborts (see "Merge Procedure Performance Is Slow or Unstable" on page 125).

For more information about the merge procedure, see the chapter about stored procedures in the *Interaction Concentrator 7.6 User's Guide.*

Merge Procedure Does Not Complete Successfully

There are a number of possible causes of this problem, and describing them is beyond the scope of this section.

If the merge procedure does not proceed to the completed state, in addition to the steps you must take to resolve the specific cause of the merge procedure failure, you might also have to reset the merge procedure so that it recovers from its failed state (see "Merge Procedure Recovery").

Note: Under some circumstances, merge procedure recovery is not required. For example, the merge procedure may fail to complete successfully as a result of a deadlock condition. In this case, no special action is required other than to run the procedure again. However, if an error is discovered in the merge procedure, or more than one instance of ICON is running concurrently and the procedure state becomes corrupted, then the merge procedure recovery should be performed.

The following tables store information about the state of the merge procedure:

- `GSYS_PENDING_IR`
- `GSYS_PENDING_LINK`
- `GSYS_SYSPROCINFO`

Procedure: Merge Procedure Recovery

Purpose: To reset the merge procedure to recover from a failed state.

Start of procedure

1. Ensure that the merge procedure (`gsysIRMerge` or `gsysIRMerge2`) is not active.
2. Truncate the `GSYS_PENDING_IR` and `GSYS_PENDING_LINK` tables.
3. In the `GSYS_SYSPROCINFO` table, delete the rows in which `PROCNAME = 'gsysirmerge'`.
4. Reset the database lock if necessary (see Resetting the Database Lock Manually).

End of procedure

Merge Procedure Does Not Execute

Possible causes of this problem are as follows:

- The stored procedure was called incorrectly.

  Solution: Verify the syntax of the call to execute `gsysIRMerge` or `gsysIRMerge2`, and correct as required. For more information, see the chapter about stored procedures in the *Interaction Concentrator 7.6 User's Guide.*

- The database lock was not released after a previous execution (or attempted execution) of the merge procedure.

  The merge procedure sets a lock on IDB to prevent concurrent execution of the procedure and resulting data corruption. Steps to identify and correct the problem of an unreleased lock depend on whether the procedure was invoked by a call to `gsysIRMerge` or directly to `gsysIRMerge2`.

  - `gsysIRMerge` was called—`gsysIRMerge` runs `gsysIRMerge2` with the lock override setting at its most conservative (never override). If the previous lock was not released, the merge procedure aborts after doing the following:
    - Creates a record in the `G_LOG_MESSAGES` table with `MESSAGE_ID = 5020` and `MESSAGETEXT = 'LOCKED'`.
    - Creates a corresponding record in the `G_LOG_ATTRS` table with attributes `'PREVCALLER'` (the name of the previous caller) and `'PREVAGE'` (the number of seconds since the lock was obtained by the previous caller).

    Solution: Release the lock, then call `gsysIRMerge` again. To release the lock, see Resetting the Database Lock Manually.

  - `gsysIRMerge2` was called directly—`gsysIRMerge2` provides a `RESULT` output parameter to assist you in diagnosing the problem. If `RESULT==2`, the procedure failed to execute because the database was locked.

    Solution: Call `gsysIRMerge2` again, specifying an appropriate value for the `OVERRIDE` input parameter, and other input parameters as applicable, to allow this execution of the stored procedure to override the database lock. For more information about `gsysIRMerge2` parameters, see the chapter about stored procedures in the *Interaction Concentrator 7.6 User's Guide.*

Procedure: Resetting the Database Lock Manually

Purpose: To manually release the lock on IDB introduced by the `gsysIRMerge` stored procedure.

Start of procedure

1. Ensure that the stored procedure is not running. The steps to verify this depend on your deployment and type of RDBMS.
2. Issue the following statement (the exact syntax depends on the RDBMS):

   ```
   update GSYS_SYSPROCINFO
   set TRANSACTION_ID = 0
   where DOMAINID = 0
   and  PRIMARYID = 99
   and  PROVIDERTAG = 0
   and  PROCNAME = '<CALLER>'
   ```

   <CALLER> is the value of the CALLER parameter that was provided to gsysIRMerge2 in the procedure call that resulted in the lock. If gsysIRMerge2 was called by gsysIRMerge, the CALLER was 'singleIDBMerge'.
3. Commit the statement issued in Step 2.

End of procedure

Merge Procedure Performance Is Slow or Unstable

Possible causes of this problem are as follows:

- On a DB2 platform, default values of certain database parameters result in an excessive number of deadlocks.

  Solution: Significantly increase the values of the LOCKLIST and MAXLOCKS database configuration parameters.

  Alternatively, execute ALTER TABLE...LOCKSIZE TABLE statements against the G_IR, G_CALL, and G_IS_LINK tables.
- On a Microsoft SQL platform, in large-scale deployments, default values of certain merge procedure parameters are not optimal.

  Solution: Significantly increase the values of the step and limit parameters in the G_DB_PARAMETERS table. For more information, see the chapter about stored procedures in the *Interaction Concentrator 7.6 User's Guide.*

Index

A

C

D

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

U

V

W

X