Interaction Concentrator Deployment Guide

Interaction Concentrator 8.1.5

12/30/2021

Table of Contents

ICON Deployment Guide **4**
Document Change History **6**
New Features and Modifications in 8.1.5 **15**
About Interaction Concentrator **35**
The Interaction Concentrator Components and What They Do 37
Supported Deployment Scenarios 42
ICON Roles 59
Planning Your Deployment **62**
Configuration and Installation **70**
Deploying DB Server 72
Deploying ICON 73
Deploying IDB 81
Deploying DAP 88
Special Configuration Requirements **90**
Configuring for Voice Data 91
Configuring for Multimedia Data 95
Configuring for Attached Data 97
Configuring for Virtual Queue Data 105
Configuring for Agent State and Login Data 108
Configuring for Outbound Contact Data 112
Configuring for LRM Data 117
Configuring a Partitioned Oracle IDB 119
Configuring for High Availability 122
Configuring for Multi-Language Support 123
Configuration Options **126**
Interaction Concentrator Options 127
ICON Configuration Options, <nowiki>[</nowiki>callconcentrator<nowiki>]</nowiki> Section 128
Switch Options 129
DN Options 130
Script Options 131
DAP Option 132
Starting and Stopping **133**
Troubleshooting **140**
Attached Data Specification File **149**

Sample Script for Custom Attached Data 156
Appendix: Log Events 159
Appendix: Migration Procedures 171

ICON Deployment Guide

What is Interaction Concentrator?
Genesys Interaction Concentrator collects and stores detailed data from various sources (such as T-Server, Interaction Server, and Configuration Server) in a contact center that is empowered with Genesys software. Downstream reporting systems can access Interaction Concentrator data in near real time.

This guide includes the following information:

Features and Functionality

Overview of ICON architecture, components, and features.

Features and Functionality

New Features and Modifications in 8.1.5

Components and Their Functions—Introduces the ICON Server and IDB.

Planning Your Deployment

Prerequisites and Considerations lists compatible Genesys components, prerequisites, and deployment considerations.

This section also includes information about supported deployment scenarios and the role option.

Configuring and Installing

Configuring and Installing includes deployment instructions.

Special Configuration Requirements provides additional configuration needed for various types of deployments (such as multimedia, Outbound, and LRM)

ICON configuration options.

Special Appendixes

Appendix: Log Events, which contains all log events created or updated in 8.1.5 releases.

Appendix: Migration Procedures, which documents the procedure for migrating to 8.1.5 releases.

Additional Information

Troubleshooting

Starting and Stopping

Attached Data Samples—Attached Data Specification File and Sample Script for Custom Attached Data

Document Change History

This topic provides a record of documentation improvements and corrections made outside the scope of a software release.

This topic also provides a historical record of document content that was new or that changed significantly since the initial 8.1 release of Interaction Concentrator.

The most recent changes appear first.

> **Important**
>
> - There was no 8.1.3 release of Interaction Concentrator.
> - There was no 8.1.513 release of Interaction Concentrator.

New in Document Version 8.1.514.47

- Information about an issue that might cause the merge and purge procedures to fail in MS SQL Server 2019 Cluster deployments has been added to the Troubleshooting page.

Change in Document Version 8.1.514.08

- Updated the description of the **[callconcentrator]**:acc-proc-tout option to indicate that, in all all 8.x releases, this option has a hard-coded value of 1 second and does not require you to set a value. If you change the option value, Interaction Concentrator disregards it.

New in Document Version 8.1.514.06

- To restore backward compatibility, the default value for the **max-userdata-length** option has been changed from 1024 to 255. For details, see the **[callconcentrator]**:max-userdata-length option description.
- The topic containing the tables of new features and other changes that were made to Interaction Concentrator in 8.1.5 releases, which was part of the Interaction Concentrator online migration document, has been moved the New Features and Modification in 8.1.5 topic in the *Deployment Guide*.

It has also been reformatted to provide a more complete list of new features, with links to the major documentation updates made for each new feature or modification.

- The migration procedures specific to 8.1.5 releases have been moved from the online migration document to the Appendix: Migration Procedures topic in the *Deployment Guide*. Note that this topic contains migration information only for releases 8.1.5 and higher. For instructions on how to migrate to earlier releases of Interaction Concentrator, see the Interaction Concentrator chapters in the Genesys Migration Guide.

New in Document Version 8.1.514.03

- To support the addition of functionality enabling you to set an alarm for situations in which Interaction Concentrator call handling rules result in unprocessed or destroyed calls, added descriptions of the new configuration option, **[callconcentrator]**:log-call-failure, and the new log event on which you can set an alarm, 09-20039. For more details on this functionality, see Setting Alarms for Call Processing Failures in the *Interaction Concentrator User's Guide*.
- Added information specifying the release starting with which Genesys Info Mart can store voice and multimedia data in the same IDB.
- Updated the description of the Configuration information ICON stores to indicate that ICON always stores data for all Tenants in your environment.
- Removed references to the **send_attribute** configuration option, configured on the **Field** object, from Configuring for Outbound Contact Data. This option is not required for reporting on Outbound Contact interactions.
- The texts of two log events, 09-25032 and 09-25033, are revised for clarity.

New in Document Version 8.1.514.02

- Interaction Concentrator now stores a user-configurable number of the last calls/interactions and parties associated with a device. This enables more detailed reporting on the progress and outcome of interactions. To support this functionality, the max-party-info configuration option has been added to the **[custom-states]** section.

New in Document Version 8.1.512.00

- The new **[callconcentrator]**:timestamp-processing configuration option replaces the recently-introduced **[callconcentrator]**:ignore-milliseconds option. The **timestamp-processing** option offers a more precise specification of how you want ICON to handle any necessary shortening of millisecond and microsecond timestamp values.
- To keep backward compatibility with environments running Genesys Info Mart 7.6 or Genesys Info Mart 8.x releases earlier than 8.5.007.14, the new **[callconcentrator]**:max-userdata-length configuration option enables you to set the maximum data length in certain fields to 255 characters in IDB data.

- Added a note to the description of the **[callconcentrator]**:adata-default-storage configuration option indicating that the values of the **attr_is_online** and **attr_itx_agent_id** attributes are always stored in the G_USERDATA_HISTORY table.
- Added a note to the Switch-level same-dn option description and to the Recognizing the Correct DN in Environments Where Internal and External DNs Have the Same Name section explaining that ICON does not correctly construct an external DN if it has the same name as the internal DN of the originating party.
- Added a section in the Connections toggle section on the Configuring for Outbound Contact Data page explaining how to add OCS and T-Server to your ICON `Application` object **Connections** tab so as to gather data correctly from the desired switches.
- Support for use of the HTTP Listener to monitor and report on Interaction Concentrator performance has been discontinued as of this release. A note recording the end of support has been added to the "HTTP Listener" section under the Configure the Options tab heading. As a result of this discontinued support, the following configuration sections and options are obsolete for release 8.1.512.08 and later:
  - **[listeners]** section
  - **[user_named_section]** section (by default, this section is known as the **[http]** section)
  - user-named-option configuration option in the **[listeners]** section
  - port configuration option in the **[user_named_section]** section
  - protocol configuration option in the **[user_named_section]** section
  - transport configuration option in the **[user_named_section]** section

New in Document Version 8.1.511.00

- Added a description of the new **[callconcentrator]**:ignore-milliseconds configuration option, which enables you to choose whether you want Interaction Concentrator to truncate microsecond values to seconds or round off microsecond values to milliseconds before storing the values in IDB.
- Updated the description of the **[callconcentrator]**:ssc-processing configuration option to specify that Interaction Concentrator does not support single-step conferences that include an external DN.
- Added notes indicating that PostgreSQL IDB requires you to use DB Server release 8.1.301.11 or higher, which uses DB Client 8.5.1, to the Planning your Deployment and Deploying DB Server pages.
- Consolidated the deployment information by moving the content from the top-level Deployment Procedures page (now deleted) to the top of the Configuration and Installation page.

New in Document Version 8.1.510.00

- Added the following information relevant to Unicode support for IDB on Microsoft SQL (Unicode is already supported for Oracle and PostgreSQL RDBMSs; it is not available for DB2):
  - Notes about Unicode support in the following locations:
    Compatibility.

Deploying DB Server.

Deploying IDB.

- A new section, Configuring for Unicode Support in an Environment with a Microsoft SQL IDB, which includes full deployment instructions. This section is in a new topic, Configuring for Multi-Language Support, which contains information collected from previously-scattered locations in the `Deployment Guide` and `User's Guide`.
- The description of a new configuration option, **[callconcentrator]**:support-unicode.
- Descriptions of five new log events, 09-25032, 09-25033, 09-25034, 09-25035, and 09-25036.

- Moved the information formerly contained on the Special Configuration Requirements topic to separate subtopics to make it easier to locate information.
- Moved two configuration-related sections from the How ICON Works page in the *User's Guide* to the *Deployment Guide*: Configuring for Multi-Language Support and Configuring Conferencing and Transfer Options.
- Added a note under "Interaction Database" to the deployment planning information recommending that DB Server be located close to a PostgreSQL IDB.
- Added a note to the **[callconcentrator]**:use-dss-monitor and **[callconcentrator]**:role configuration options indicating that if the **use-dss-monitor** option is not configured, or is set to `false` or `0`, the GSYS_DOMAIN field in role-related IDB tables contains the value `0` (zero).

New in Document Version 8.1.509.00

- Added a note in the description of the **[callconcentrator]**:use-server-partyuuid configuration option explaining that the parent parties identified by ICON and SIP Server might not correspond.

New in Document Version 8.1.508.00

- Added a short explanation of the new functionality that enables ICON to correctly store the value for otherDN when both an internal (local) DN and an external DN have the same name. This functionality requires you to configure the new Switch-level same-dn configuration option.
- Added a description of the new **[callconcentrator]**:use-server-partyuuid configuration option, which enables Interaction Concentrator to support multiple routing scenarios in single-site and multi-site environments running SIP Server release 8.1.102.13 or higher.
- Added information about the settings required for internationalization on the PostgreSQL RDBMS.
- Updated a number of ICON configuration options to indicate that the values `0` and `1` are valid as well as the already-documented values of `false` and `true`.

New in Document Version 8.1.507.00

- Updated the descriptions of the **[custom-states]**:store-event-data and **[custom-states]**:EventData options to indicate the settings required to report on EventCustomReporting data (which can be used to report on interaction focus time).
- Created a Log Events Appendix, which contains full documentation of all Interaction Concentrator log events that have been added since the latest publication of the Framework Combined Log Events Help.
- Corrected the list of user data keys that must be explicitly specified in the attached data specification file to capture all possible Universal Routing Server (URS) attached data. For details, see Universal Routing Server Attached Data.
- Updated the description of the DN-level ring-divert option to indicate that ICON can identify the PARENTPARTYID and the PARENTLINKTYPE of the Ringing party even if the **ring-divert** option is set to `0`, as long as the Switch-level lookup-queue-on-ringing option is set to 1.

New in Document Version 8.1.506.00

- ICON now enables you to run multiple ICON Applications from the same directory using the same executable. See Running Multiple ICON Applications from the Same Directory for details. The new **[callconcentrator]**:cfg-dbname option must be configured to support this functionality.
- The description of the **[callconcentrator]**:store-releasing-party configuration option has been updated to specify that ICON supports this functionality for all switches that provide the necessary information. To determine whether your switch supports this functionality, check your T-Server/SIP Server documentation.
- Added a recommendation to set the NLS_LANG environment variable in all installations using an Oracle database.

New in Document Version 8.1.505.00

- ICON now hides sensitive attached data information in the ICON log file at every log level. For details, see the Security Features section in Prerequisites and Hide Selected Data in Logs (in the *Genesys Security Deployment Guide*).
- Added information about the new **Purge2_PartitionType0_ora.sql** database initialization script, which optimizes performance in environments with large Oracle-based IDBs in which the ICON **[callconcentrator]**:partition-type option is set to `0` (the default value).

New in Document Version 8.1.504.00

- Added a brief statement of support for Hunt Groups, with links to the relevant configuration information.

- The valid value for minimum re-registration timeout in the Switch-level gcti-re-registration-tout configuration option has been corrected. `0` is now included as a valid value.

New in Document Version 8.1.503.00

- Added information on the new DN re-registration functionality. For details see "Configuring DN Re-registration" on the Configuring for Voice Data page and the description of the new gcti-re-registration-tout Switch-level configuration option.
- Added a note to inform you that the separate purge procedures (gsysPurgeIR, gsysPurgeUDH, gsysPurgeLS, and, gsysPurgeOS) have been discontinued as of this release. To purge IDB, use the gysyPurge81 or purgePartitions811 purge procedure.
- Corrected the descriptions for the delivered-flag, support-dn-type-N, and suppress-user-data Switch-level configuration options.

New in Document Version 8.1.502.00

- Edited the instructions for adding a connection to Interaction Server to indicate that you can now add an Interaction Server type Application object to the ICON **Connections** tab. Previously, you had to add a connection to an Application object of the type T-Server, which was a stand-in for the Interaction Server Application object. For the instructions on how to configure the T-Server connection required by previous releases, see Configuring for Multimedia Data.
- Added a new section—MultimediaDeployments—to the [[Documentation:ICON:Dep:Architecture|Supported Deployment Scenarios page, describing the support for connections to multiple Interaction Servers.
- Edited descriptions of the following configuration options:
  - x-print-attached-data (in the ICON **[log]** section)
  - The Switch-level **support-dn-type-5** option has been renamed to support-dn-type-N. **support-dn-type-5** is just one of the valid names for this option, in which the variable *N* should be replaced by the number indicating the DN type to which the option setting will apply.
  - Corrected the valid values for the Switch-level call-deletion-timeout option.
  - Corrected the default value for the Switch-level emulate-event-queued-extrp, emulate-event-queued-rp, and emulate-event-queued-rq options.
- Removed the following obsolete configuration options:
  - min-tsync-roundtrip (formerly configured on the Switch configuration object)
  - tsync-threshold (formerly configured on the ICON `Application` object)

New in Document Version 8.1.501.00

- Added descriptions of the following configuration options:
  - **[callconcentrator]**:ssc-processing—Added in release 8.1.500.04.
  - lookup-queue-on-ringing (Switch-level)—Added in release 8.1.400.20.
  - **[callconcentrator]**:use-nts-call-state—Added in release 8.1.400.14.
- Added a note about support for single-step conference scenarios to the description of the DN-level ring-divert option.
- Revised the information about the attached data specification file to indicate that changes to the file are now made dynamically. See Configuring for Attached Data and the **About** tab on the Attached Data Specification File page.
- Updated the configuration recommendations for License Reporting Manager (LRM) to include the updated instructions on how to configure the ICON `lrm` role for LRM reporting. See the **Recommended Role Assignments** tab on the ICON Roles page and Configuring for LRM Data.

New in Document Version 8.1.401.00

- The *Deployment Guide* was converted from PDF-only format to this online format, with the option of printing a PDF, if desired.
- Added descriptions of two new configuration options: **[callconcentrator]**:cfg-annex and the DN-level do-not-register.
- Updated the name of the new migration script name format used in releases 8.1.4 and higher. For details on the migration scripts, see the *Interaction Concentrator 8.1 Deployment Procedure*.
- Added an item to the **About Interaction Concentrator** topic noting support for encrypted RDBMSs and support for having TEvent attached data hidden in log files. For details on this functionality and how to configure it, see the *Genesys Security Deployment Guide*.

New in Document Version 8.1.201.00

- Added information on how to configuring IDB on PostgreSQL and specified the details of DAP configuration appropriate to PostgreSQL.
- Added notes stating new ICON support for dynamic updates to T-Server host and port changes.
- Added a section on security features and configuration requirements to the section on Deployment Planning.
- Added a recommendation to install ICON on the same host as the T-Server or Interaction Server for which it writes data.
- Added information about the option to purge the persistent queue file upon ICON startup to the information on the Persistent Queue file.
- Added notes to indicate that the **gsysIRMerge** and **gsysIRMerge2** stored procedures are compatible

with Genesys Info Mart 7.x but are not needed by Genesys Info Mart 8.x.

- Added notes to indicate that the **gsysIRMerge** and **gsysIRMerge2** stored procedures are not applicable to the PostgreSQL RBDMS.
- Configuration option additions, updates and corrections:
  - Added descriptions of the **[callconcentrator]**:cseq-adjustment, **[callconcentrator]**:gls-enforce-reason-code, **[callconcentrator]**:pq-startup-purge, ring-divert (Switch-level and DN-level), and gls-improve-data-for-agent (Switch-level) configuration options.
  - Added a reference to the information about the `GSYS_EXT_INT1` field in the `G_PARTY_HISTORY` table to the description of the DN-level ivr configuration option.
  - Added notes to the **[callconcentrator]**:om-memory-optimization and **[callconcentrator]**:calls-in-the-past options stating the releases in which the **om-memory-optimization** setting could override the **calls-in-the-past** setting.
  - Corrected and clarified the wording for the gls-acw-first option (configured on the ICON `Application` object, **[callconcentrator]** section, and the `Switch` object).
  - Corrected and amplified the description for the Switch-level sst-options configuration option.
  - Added notes to advise users that the values for the **[callconcentrator]**:role option must be set in lowercase only, not uppercase or mixed case.
- Updated the name of the Genesys eServices product by changing it from eServices/Multimedia simply to eServices, its current name. Multimedia was the former product name.
- Added notes instructing readers to restart the ICON server after any change to the attached data specification file.

New in Document Version 8.1.101.00

- Purge by Truncating Partitions
  - Described new scripts and stored procedure, **purgePartitions811**, for creating and using a partitioned Oracle RDBMS that is set up so as to be able to purge data by truncating partitions:
  - Added the **purgePartitions811** stored procedure to the list of purge procedures.
  - Noted important considerations to take into account before implementing the purge by truncating partitions functionality.
  - Added deployment instructions for creating a partitioned Oracle IDB that can then be purged using the **purgePartitions811** stored procedure.
  - Listed the new initialization scripts used to partition IDB and to enable purging by truncating partitions.

> ### Important
> In this release, partitioning is supported only on Oracle 11 or higher.

- New Configuration Options—Added descriptions of the following configuration options:

    - **[callconcentrator]**:cfg-auto-resync
    - **[callconcentrator]**:advanced-ext-party-reconstruction
    - **[callconcentrator]**:dest-busy-processing
    - **[log]**:x-server-http-trace-level
    - **[log]**:x-server-smtp-trace-level
    - **[log]**:x-server-dbw-debug-level
    - Switch-level third-party-queue-in-divert
- Corrected Configuration Option Values
    - Corrected the Changes Take Effect values for the **[callconcentrator]**:pq-backlog-alarm-threshold and **[callconcentrator]**:pq-backlog-clearance-threshold options from `Immediately` to `After restart`.
- Added Information on IDB Update Scripts for Genesys Info Mart—Added descriptions of the **update_idb_for_gim.sql** and **update_idb_for_gim_mm.sql** scripts, required for environments running Genesys Info Mart.
- New PQ File Requirement—Added a note informing users of the requirement to locate persistent queue files on a local drive.

New Features and Modifications in 8.1.5

This topic lists changes (additions, modifications, and discontinuations) in the product made during the 8.1.5 release timeframe that might affect an existing deployment. If a change required documentation updates, this topic links to the updated documentation.

- For a comprehensive list of changes in each release, including corrections, review the "New in This Release" sections of the Interaction Concentrator 8.1 Release Note.
- For updates and corrections to the documentation that were not triggered by a new feature, improvement, or discontinued support, see the Document Change History topic in each book.
- This topic pertains only to release 8.1.5. For a similar list of changes that might affect your deployment in earlier releases of Interaction Concentrator, see the Interaction Concentrator chapters in the *Genesys Migration Guide*.

This topic contains the following sections:

- General List of New Features and Modifications
- Changes to Configuration Options
- Changes to the Interaction Database (IDB)

General List of New Features and Modifications

The following significant changes in functionality were made in 8.1.5 releases.

Feature Description	Type of Change	Occurred in Release	Documentation Updates
You can now have Interaction Concentrator update some fields in the GM_F_USERDATA table that initially contained NULL values with new data received in the EventPropertiesChanged event. The new **[callconcentrator]**:update-ixn-f-adata configuration option, which is set by default to `false`, enables this functionality.	Improvement	8.1.514.12 (05/16/2018)	*Options Reference:* • **[callconcentrator]**:update-ixn-f-adata
Interaction Concentrator now correctly supports two-step transfer and conference scenarios in SIP Cluster deployments. An agent invited into a consultation call and then moved to the main call now has the READY state after	Improvement	8.1.514.10 (11/30/2017)	

Feature Description	Type of Change	Occurred in Release	Documentation Updates
end of the call. Previously, the agent continued to be shown as BUSY.			
The new **[callconcentrator]**:trim-broken-utf8 configuration option can enable ICON to check for and remove incomplete UTF-8 symbols from the end of a truncated user-data string. These incomplete UTF-8 symbols might occur when a long user-data string is truncated in the middle of a multi-byte UTF-8 character.	Improvement	8.1.514.10 (11/30/2017)	*Options Reference:* • **[callconcentrator]**:trim-broken-utf8
Interaction Concentrator now records the data necessary to report on supervision (monitoring) of agent calls. The relevant data is provided in AttributeExtensions of EventUserEvent and is stored in the G_CUSTOM_DATA_S table. Three new options have been introduced to support this functionality.	New Feature	8.1.514.09 (10/31/2017)	*Options Reference:* • **[callconcentrator]**:enable-supervision-subscription • **[custom-states]**:store-event-extensions • **[custom-states]**:EventExtensions
The following new indexes were added to ICON schema: • IDX_G_AGENT_STATE_H_PID_TYPE • IDX_G_AGENT_STATE_H_LSID_LSEQ • IDX_G_PARTY_GE_INT_CID	Improvement	8.1.514.08 (09/26/2017)	See the Index table for the following schema tables in the *Physical Data Model Reference* for your RDBMS: • G_AGENT_STATE_HISTORY • G_PARTY
In all 8.x releases of Interaction Concentrator, the **[callconcentrator]**:acc-proc-tout option has a hard-coded value of 1 second and does not require you to set a value. If you change the option value, Interaction Concentrator disregards it. In releases up to 8.1.514.08, this option was incorrectly described as having a configurable value.	Correction	8.x	*Options Reference:* • **[callconcentrator]**:acc-proc-tout
Four configuration options have been adjusted to support dynamic changes to the values, which is consistent with the documentation.	Improvement	8.1.514.08 (09/26/2017)	*Options Reference:* • **[callconcentrator]**:acc-queue-lifespan • **[callconcentrator]**:acc-queue-size

Feature Description	Type of Change	Occurred in Release	Documentation Updates
			• **[callconcentrator]**:pq-purge-number • **[callconcentrator]**:dbw-seq-tout
To restore backward compatibility, the default value for the **[callconcentrator]**:max-userdata-length option has been changed from 1024 to 255.	Improvement	8.1.514.06 (08/07/2017)	*Options Reference:* • **[callconcentrator]**:max-userdata-length
Added the ability to set an alarm when calls are being destroyed due to the way Interaction Concentrator handles problems with missing call data	New functionality	8.1.514.03 (05/02/2017)	*Options Reference:* • **[callconcentrator]**:log-call-failure *Deployment Guide:* • The new log event on which you can set an alarm, 09-20039. *User's Guide:* • Setting Alarms for Call Processing Failures.
You can now store a user-configurable number (from 4 to 128) of last calls/interactions and parties associated with a device. Previously, this value was always 4; the new default value is 16. Recording a larger number of last calls/interactions and parties that are supplied in EventUserEvent and EventCustomReporting events enables more detailed reporting on the progress and outcome of interactions. To support this functionality, the **[custom-states]**:max-party-info configuration option has been added. Note that this change does not affect the IDB schema, only the values recorded in the specified fields.	New functionality	8.1.514.02 (04/14/2017)	*User's Guide:* • How to configure the number of last calls/interactions and parties associated with a device that should be stored in the G_CUSTOM_DATA_P, G_CUSTOM_DATA_S, and G_CUSTOM_STATES tables. See Custom States in Interaction Concentrator. *Options Reference:* • **[custom-states]**:max-party-info
Improved handling of timestamps having fractions of milliseconds in TEvents and Interaction Server events. To accomplish this, the **[callconcentrator]**:timestamp-processing option replaces the	Improvement	8.1.512.08 (01/31/2017)	*Options Reference:* • **[callconcentrator]**:timestamp-processing

Feature Description	Type of Change	Occurred in Release	Documentation Updates
now-deprecated **[callconcentrator]**:ignore-milliseconds configuration option.			• **[callconcentrator]**:ignore-milliseconds
To keep backward compatibility with environments running Genesys Info Mart 7.6 or Genesys Info Mart 8.x releases earlier than 8.5.007.14, the new **[callconcentrator]**:max-userdata-length configuration option enables you to set the maximum data length in certain fields to 255 characters in IDB data.	New functionality	8.1.512.08 (01/31/2017)	*Options Reference:* • **[callconcentrator]**:max-userdata-length
Discontinued support for use of the HTTP Listener to monitor and report on Interaction Concentrator performance.	Discontinued functionality	8.1.512.08 (01/31/2017)	*Deployment Guide:* • Numerous updates. For specifics, see the Document Change History topic. *User's Guide:* • Monitoring Interaction Concentrator.
The length of certain fields was changed to better support the requirements for the data stored in those fields. For details, see the Document Change History topic in the *Physical Data Model* document for your RDBMS.	Changed datatypes in some fields	8.1.512.08 (01/31/2017)	Physical Data Model documents: • For specifics, see the Document Change History topic in the Physical Data Model document for your RDBMS.
Interaction Concentrator now enables you to choose whether microsecond values that are stored in IDB are truncated to seconds or rounded to milliseconds. Note that ICON supports a precision-level of seconds, not milliseconds, for time values.	New functionality	8.1.511.01 (10/31/2016)	*Options Reference:* • **[callconcentrator]**:ignore-milliseconds
Added support for Microsoft SQL Server 2014.	New functionality	8.1.511.01 (10/31/2016)	

Interaction Concentrator now supports Unicode data on Microsoft SQL RDBMSs, as well as Oracle and PostgreSQL RDBMSs. Unicode encoding is not supported on DB2 RDBMSs. To support this functionality, a new configuration option, **[callconcentrator]**:support-unicode, has been added and ICON performs additional tests to check	New functionality	8.1.510.07 (09/23/2016)	*Deployment Guide:*
			• The following new log events:
			  • 09-25032
			  • 09-25033
			  • 09-25034
			  • 09-25035

Feature Description	Type of Change	Occurred in Release	Documentation Updates
whether the encoding settings are consistent in your environment. The results of these checks are reported in five new Standard-level log events.			• 09-25036. • A new section, Configuring for Unicode Support in an Environment with a Microsoft SQL IDB, which includes full deployment instructions. This section is in a new topic, Configuring for Multi-Language Support, which contains information collected from previously-scattered locations in the *Deployment Guide* and *User's Guide*. *Options Reference:* • **[callconcentrator]**:support-unicode
Interaction Concentrator now supports the Interaction Server ability to distinguish the agent login region.	New functionality	8.1.510.07 (09/23/2016)	
ICON can now distinguish the correct type (Internal or External) of otherDN when both an internal DN and an external DN have the same name, enabled by the Switch-level same-dn configuration option.	New functionality	8.1.508.09 (04/22/2016)	*Deployment Guide:* • The new Recognizing the Correct DN in Environments Where Internal and External DNs Have the Same Name section. *Options Reference:* • same-dn
Interaction Concentrator can now support multiple routing scenarios in single-site and multi-site environments running SIP Server release 8.1.102.13 or higher. To support this functionality, a column, TS_PARENTPARTYGUID, has been added to the G_PARTY table, and the **[callconcentrator]**:use-server-partyuuid configuration option has been added.	New functionality	8.1.508.09 (04/22/2016)	*Options Warehouse:* • **[callconcentrator]**:use-server-partyuuid
Interaction Concentrator improved handling of stuck calls, login sessions, agent states, reason codes, and virtual queue records, enabling downstream reporting	New functionality	8.1.508.09 (04/22/2016)	

Feature Description	Type of Change	Occurred in Release	Documentation Updates
applications, such as Genesys Info Mart, to improve their interaction reporting.			
ICON now stores data that enables reporting on agent focus time. The data comes in the form of key-value pairs (KVPs) in the **attr_event_content** attribute in the EventCustomReporting event, which is passed to ICON via Interaction Server. The data provides information for reporting on focus time—that is, how long a particular interaction was in focus (actively being processed) on the agent desktop.	New functionality	8.1.507.07 (02/09/2016)	*User's Guide*: • The new Processing Data for EventCustomReporting section. *Options Reference:* • **[custom-states]**:store-event-data • **[custom-states]**:EventData
ICON now stores data provided by Chat Server that enables you to determine who ended a chat session.	New functionality	8.1.507.07 (02/09/2016)	*User's Guide:* • The new Chat Session Attributes that Indicate Who Ended the Session section.
Ability to change the name and location of the **cfg-sync.db** file, which enables you to run multiple ICON Applications from the same directory using the same executable, enabled by the new **[callconcentrator]**:cfg-dbname option.	New functionality	8.1.506.07 (12/04/2015)	*Deployment Guide:* • The new Running Multiple ICON Applications from the Same Directory section. *Options Reference:* • **[callconcentrator]**:cfg-dbname
Hiding of sensitive attached data information in the ICON log file at every log level, if configured to do so. ICON has enabled you to hide sensitive attached data information for TEvents since release 8.1.2. With this release, ICON now enables you to hide such information for SQL statements as well.	New functionality	8.1.505.05 (10/19/2015)	*Deployment Guide*: • Updated Security Features section in Prerequisites. *User's Guide*: • Updated Security Features section. *Genesys Security Deployment Guide*: • Updated Hide Selected Data in Logs.
Support for both sequential and parallel Hunt Group calls controlled by Genesys SIP Server (supported only in standalone SIP Server environments).	New functionality	8.1.504.04 (08/03/2015)	*Deployment Guide:* • The new support for Hunt Groups section, with links to relevant configuration

Feature Description	Type of Change	Occurred in Release	Documentation Updates
			information.
Mutual TLS support	New functionality	8.1.504.04 (08/03/2015)	*Genesys Security Deployment Guide:* • Updated the Mutual TLS section.
Automatic re-registration of unregistered DNs, enabled by the new Switch-level gcti-re-registration-tout option. This functionality is available on voice Switches only.	New functionality	8.1.503.03 (06/09/2015)	*Deployment Guide:* • The new "Configuring DN Re-registration" section in the Configuring for Voice Data topic. *Options Reference:* • gcti-re-registration-tout
Improved purge locking. The purge procedure now uses a more robust native RDBMS approach and avoids the previously encountered issue of multiple purge processes running in parallel.	New functionality	8.1.503.03 (06/09/2015)	*User's Guide:* • The new Purge Lock Mechanisms section.
Microsoft SQL 2012 Cluster	Added support	8.1.503.03 (06/09/2015)	
Added support for direct connections to applications of type Interaction Server. This enables support for multiple Interaction Servers per tenant and for multi-tenant Interaction Servers. • If you have connections to Interaction Server that were created using a specially-configured T-Server application, you can continue using them. You do not need to change your existing connections to Interaction Server.	New functionality	8.1.502.04	*Deployment Guide:* • The updated instructions for adding a connection to Interaction Server section. • The updated Multimedia Deployments section. The *Physical Data Model* documents: • Added value in certain IDB tables. For details, see Interaction Database (IDB).
The separate purge procedures (gsysPurgeIR, gsysPurgeUDH, gsysPurgeLS, and, gsysPurgeOS) have been discontinued. To purge IDB, use the gysyPurge81 or purgePartitions811 purge procedure.	Discontinued support	8.1.502.04	*Deployment Guide:* • Added a note to inform you of the discontinued support. *User's Guide*: • Added a note to the Using

Feature Description	Type of Change	Occurred in Release	Documentation Updates
			Separate Purge Procedures section.
IBM DB2 v10	Added support	8.1.502.04	
Added support for single-step conference scenarios, enabled by the new **[callconcentrator]**:ssc-processing configuration option.	New functionality	8.1.500.04	The *Physical Data Model* documents: • Changes in the data written in certain IDB tables. For details, see Interaction Database (IDB). *Options Reference:* • **[callconcentrator]**:ssc-processing
Interaction Concentrator now supports recognition by Genesys Info Mart of IWS chat conference scenarios (*visibility mode*). To provide this support, ICON records a value for the eServices **attr_visibility_mode** interaction attribute in the keyname field of the G_USERDATA_HISTORY table. The attribute value indicates the mode of an agent who has joined a (chat) conference: 1 (Conference mode), 2 (Monitor mode), or 3 (Coach mode, also known as *chat consult*).	New functionality	8.1.500.04	
Interaction Concentrator can now dynamically change the list of user data key names and their definitions, enabling you to avoid restarting ICON. In addition, you can now store the same key name multiple times as long as it is stored in different tables or comes from different data sources. Requires you to configure the new **[callconcentrator]**:adata-spec-name option.	New functionality	8.1.500.04	*Deployment Guide:* • The About tab on the Attached Data Specification File topic. *User's Guide*: • The updated Processing Attached Data section. *Options Reference:* • **[callconcentrator]**:adata-spec-name
Linux 7	Added support	8.1.500.04	
Solaris/SPARC version 9 IBM AIX 5.3	Discontinued support	8.1.500.04	

Feature Description	Type of Change	Occurred in Release	Documentation Updates
Red Hat Enterprise Linux 4 HP-UX – all versions			
PostgreSQL 9.3	Added support	8.1.500.04	
IBM DB2 – all 8.x versions IBM DB2 version 9.1	Discontinued support	8.1.500.04	

Changes to Configuration Options

The following configuration options were added, removed, or changed their functionality in 8.1.5 releases. For complete documentation of all configuration options, see *Interaction Concentrator Options Reference*.

[Section] Option Name	Type of Change	Occurred in Release	Details
Interaction Concentrator Application object, **[callconcentrator]** Section • **trim-broken-utf8**	Improvement	8.1.514.10 (11/30/2017)	The new trim-broken-utf8 configuration option can enable ICON to check for and remove incomplete UTF-8 symbols from the end of a truncated user-data string. These incomplete UTF-8 symbols might occur when a long user-data string is truncated in the middle of a multi-byte UTF-8 character.
Interaction Concentrator Application object, **[callconcentrator]** Section • **enable-supervision-subscription** Interaction Concentrator Application object, **[custom-states]** Section • **store-event-extensions** • **EventExtensions**	New Feature	8.1.514.09 (10/31/2017)	Interaction Concentrator now records the data necessary to report on supervision (monitoring) of agent calls. The relevant data is provided in AttributeExtensions of EventUserEvent and is stored in the G_CUSTOM_DATA_S table. Three new options have been introduced to support this functionality: enable-supervision-subscription, store-

[Section] Option Name	Type of Change	Occurred in Release	Details
			event-extensions, and EventExtensions.
Interaction Concentrator Application object, **[callconcentrator]** Section • **acc-proc-tout**	Correction	8.x	In all 8.x releases of Interaction Concentrator, the acc-proc-tout option has a hard-coded value of 1 second and does not require you to set a value. If you change the option value, Interaction Concentrator disregards it. In releases up to 8.1.514.08, this option was incorrectly described as having a configurable value.
Interaction Concentrator Application object, **[callconcentrator]** Section • **acc-queue-lifespan** • **acc-queue-size** • **pq-purge-number** • **dbw-seq-tout**	Improvement	8.1.514.08 (09/25/2017)	The specified options have been adjusted to support dynamic changes to the values, which is consistent with the documentation. • acc-queue-lifespan • acc-queue-size • pq-purge-number • dbw-seq-tout
Interaction Concentrator Application object, **[callconcentrator]** Section • **max-userdata-length**	Changed	8.1.514.06	To restore backward compatibility, the default value for the **max-userdata-length** option has been changed from 1024 to 255. For details, see the max-userdata-length option description.
Interaction Concentrator Application object, **[callconcentrator]** Section • **log-call-failure**	Added	8.1.514.03	New option supports the ability to set an alarm to flag conditions in which calls are destroyed as a result of the way Interaction Concentrator handles missing call data. See the log-call-failure option description.
Interaction Concentrator Application object,	Added	8.1.514.02	Specifies the number of last calls/interactions

[Section] Option Name	Type of Change	Occurred in Release	Details
[callconcentrator] Section • **max-party-info**			and parties associated with a device. This enables more detailed reporting on the progress and outcome of interactions. To support this functionality, the max-party-info configuration option has been added.
Interaction Concentrator Application object, **[callconcentrator]** Section • **timestamp-processing**	Added	8.1.512.08	New option replaces ignore-milliseconds. See the timestamp-processing option description.
Interaction Concentrator Application object, **[callconcentrator]** Section • **ignore-milliseconds**	Deprecated	8.1.512.08	This option has been replaced by the timestamp-processing option.
Interaction Concentrator Application object, **[callconcentrator]** Section • **max-userdata-length**	Added	8.1.512.08	To keep backward compatibility with environments running Genesys Info Mart 7.6 or Genesys Info Mart 8.x releases earlier than 8.5.007.14, the new max-userdata-length configuration option enables you to set the maximum data length in certain fields to 255 characters in IDB data.
Interaction Concentrator Application object, **[callconcentrator]** Section • **ignore-milliseconds**	Added	8.1.511.01	Specifies whether microsecond values are truncated to seconds or rounded to achieve millisecond values. See the ignore-milliseconds option description.
Interaction Concentrator Application object, **[callconcentrator]** Section	Added	8.1.510.07	In environments using Unicode on a Microsoft SQL IDB, support-unicode indicates to Interaction Concentrator whether it should

[Section] Option Name	Type of Change	Occurred in Release	Details
• **support-unicode**			expect Unicode data and prompts it to verify that the encoding in your environment is set up consistently.
Switch object, **[gts]** Section • **same-dn**	Added	8.1.508.09	Designed to be used in environments in which internal and external DNs might have the same names. Indicates that ICON should delay party processing while waiting for Event data that can distinguish an internal DN from an external DN of the same name. See the same-dn option description.
ICON Application, **[callconcentrator]** Section and/or Switch object, **[gts]** Section • **use-server-partyuuid**	Added	8.1.508.09	Enables Interaction Concentrator to support multiple routing scenarios in single-site and multi-site environments running SIP Server release 8.1.102.13 or higher. See the use-server-partyuuid option description.
ICON Application, **[custom-states]** Section • **store-event-data** • **EventData**	Updated	Document version 8.1.507.00	Updated the descriptions of the store-event-data and EventData options to indicate the settings required to report on EventCustomReporting data (which can be used to report on interaction focus time).
DN Object, Switch Object, **Annex** tab, **[gts]** Section • **ring-divert**	Updated	Document version 8.1.507.00	Updated the description of the ring-divert option to indicate that ICON can identify the PARENTPARTYID and the PARENTLINKTYPE of the Ringing party even if the **ring-divert** option is set to 0, as long as the lookup-queue-on-ringing option is set to 1.
ICON Application,	Added	Document version	ICON now enables you

[Section] Option Name	Type of Change	Occurred in Release	Details
[callconcentrator] Section • **cfg-dbname**		8.1.506.00	to run multiple ICON Applications from the same directory using the same executable. See Running Multiple ICON Applications from the Same Directory in the **Options** tab toggle section for details. The new cfg-dbname option must be configured to support this functionality.
ICON Application, **[callconcentrator]** Section • **store-releasing-party**	Updated	Document version 8.1.506.00	The description of the store-releasing-party configuration option has been updated to specify that ICON supports this functionality for all switches that provide the necessary information. To determine whether your switch supports this functionality, check your T-Server/SIP Server documentation.
Switch object, **[gts]** Section • **gcti-re-registration-tout**	Updated	Document version 8.1.504.00	The valid value for minimum re-registration timeout in the gcti-re-registration-tout configuration option has been corrected. `0` is now included as a valid value.
Switch object, **[gts]** Section • **gcti-re-registration-tout**	Added	Document version 8.1.503.00	Added information on the new DN re-registration functionality. For details see Configuring DN Re-registration and the description of the new gcti-re-registration-tout Switch configuration option.
Switch object, **[gts]** Section • **delivered-flag** • **support-dn-type-N** • **suppress-user-**	Updated	Document version 8.1.503.00	Corrected the descriptions for the delivered-flag, support-dn-type-N, and suppress-user-data Switch configuration options.

[Section] Option Name	Type of Change	Occurred in Release	Details
data			
Switch object, **[gts]** Section • **support-dn-type-N**	Updated	Document version 8.1.502.00	Renamed from **support-dn-type-5**, which is just one of the valid names for this option, in which the variable *N* should be replaced by the number indicating the DN type to which the option setting will apply. See the support-dn-type-N option description.
ICON Application, **[log]** Section • **x-print-attached-data**	Updated	Document version 8.1.502.00	Corrected the option description. See the x-print-attached-data option description.
Switch object, **[gts]** Section • **call-deletion-timeout**	Updated	Document version 8.1.502.00	Corrected the valid values for the call-deletion-timeout option.
Switch object, **[gts]** Section • **emulate-event-queued-extrp** • **emulate-event-queued-rp** • **emulate-event-queued-rq**	Updated	Document version 8.1.502.00	Corrected the default value for the emulate-event-queued-extrp, emulate-event-queued-rp, and emulate-event-queued-rq options.
Switch object, **[gts]** Section • **min-tsync-roundtrip**	Removed	Document version 8.1.502.00	Updated the min-tsync-roundtrip option description to indicate that it is now obsolete.
ICON Application, **[callconcentrator]** Section • **tsync-threshold**	Removed	Document version 8.1.502.00	Updated the tsync-threshold option description to indicate that it is now obsolete.
ICON Application, **[callconcentrator]**	Added	8.1.500.04	Enables Interaction Concentrator to

[Section] Option Name	Type of Change	Occurred in Release	Details
Section • **ssc-processing**			provides support for single-step conference reporting. For details, see the ssc-processing option description.
Switch object, **[gts]** Section • **lookup-queue-on-ringing**	Added	8.1.400.20	Enables ICON to correctly handle scenarios having a two-step blind transfer to a Route Point in which EventRinging comes before EventDiverted. In such scenarios, the transfer is completed to a distribution DN before the call rings on the target DN. For details, see the lookup-queue-on-ringing option description.

Changes to the Interaction Database (IDB)

The following changes to the IDB schema were made in Interaction Concentrator 8.1.5. For more information about the IDB schema for Interaction Concentrator, refer to the Interaction Concentrator Physical Data Model for your RDBMS.

Feature Name	Type of Change	Occurred in Release	Details
You can now record data relating to supervisor presence and monitoring in agent calls. The relevant data is provided in AttributeExtensions of EventUserEvent and is stored in the G_CUSTOM_DATA_S table.	New Feature	8.1.514.09 (10/31/2017)	Use of this functionality requires you to have SIP Server in your environment, set the values for the SIP Server options **sip-enable-call-info** and **sip-enable-call-info-extended** to `true`, and to configure the following three new Interaction Concentrator options:enable-supervision-subscription, store-event-extensions, and EventExtensions.
Custom data KVP key names in the G_CUSTOM_DATA_S table are now truncated	Improvement	8.1.514.09 (10/31/2017)	

Feature Name	Type of Change	Occurred in Release	Details
to 64 bytes to fit into IDB fields.			
The following new indexes were added to ICON schema: • IDX_G_AGENT_STATE_H_PID_TYPE • IDX_G_AGENT_STATE_H_LSID_LSEQ • IDX_G_PARTY_GE_INT_CID	Improvement	8.1.514.08 (09/25/2017)	See the Index table for the following schema tables in the *Physical Data Model Reference* for your RDBMS: • G_AGENT_STATE_HISTORY • G_PARTY
Option replaced to improve handling of timestamps having fractions of milliseconds in TEvents and Interaction Server events.	Update	8.1.512.08	The timestamp-processing option replaces the now-deprecated ignore-milliseconds option.
To keep backward compatibility with environments running Genesys Info Mart 7.6 or Genesys Info Mart 8.x releases earlier than 8.5.007.14, you can now set the maximum data length in certain fields to 255 characters in the IDB data.	New functionality	8.1.512.08	For details, see the description of the new max-userdata-length option.
Changed datatypes in some fields	Update	8.1.512.08	The length of certain fields was changed to better support the requirements for the data stored in those fields. For details, see the Document Change History topic in the Physical Data Model document for your RDBMS.
You can now choose whether you want Interaction Concentrator to truncate microsecond values to seconds or round them to milliseconds when storing them in IDB.	New functionality	8.1.511.01	Configure this setting using the new ignore-milliseconds configuration option.
The wrapper scripts drop obsolete purge procedures.	Improvement	8.1.511.01	The following four purge procedures became obsolete in release 8.1.503.03, but they

Feature Name	Type of Change	Occurred in Release	Details
			remained in the database: gsysPurgeIR, gsysPurgeUDH, gsysPurgeLS, and gsysPurgeOS. They are now removed when you upgrade to the latest release. This change applies to Oracle, PostgreSQL, and Microsoft SQL IDBs.
Support for Unicode encoding for user data on Microsoft SQL RDBMSs, as well as Oracle and PostgreSQL RDBMSs (which already offered Unicode support). Note that Unicode is not supported for DB2 IDBs.	New functionality	8.1.510.07	To use this functionality, create a new Microsoft SQL IDB using the **CoreSchema_multilang_mssql.sql** database initialization script (rather than the **CoreSchema_mssql.sql** initialization script). This new script uses the nvarchar datatype instead of the varchar datatype to store Unicode data arriving from various data sources. For full configuration instructions and details on the functionality, see Configuring for Unicode Support in an Environment with a Microsoft SQL IDB.
Support for multiple routing scenarios in single-site and multi-site environments running SIP Server.	Column added to existing table	8.1.508.09	The new TS_PARENTPARTYGUID column in the G_PARTY table has been added to enable you to store the value associated with the SIP Server key name **parent-party-uuid** from AttributeExtensions. For details, see the description of the use-server-partyuuid option description.
Support for both sequential and parallel Hunt Group calls controlled by Genesys SIP Server.	Added values	8.1.504.04	Interaction Concentrator records the hunt group type by recording the number corresponding to the value set for the SIP Server **hg-type** configuration option.

Feature Name	Type of Change	Occurred in Release	Details
			This number is stored in the GSYS_EXT_INT1 field of the G_PARTY_HISTORY table. The numbers recorded, and the hunt group types they correspond to, are the following: • 4 = fork hunt group • 5 = linear hunt group • 6 = circular hunt group This functionality is supported only in standalone SIP Server environments. For a full description of Hunt group functionality and configuration, see Hunt Groups in Standalone Deployments in the Supplement to SIP Server Deployment Guide.
Improved purge locking	New database functionality	8.1.503.03	The purge procedure now uses a more robust native RDBMS approach and avoids the previously encountered issue of multiple purge processes running in parallel. This new database lock is used by default on Microsoft SQL and PostgreSQL databases. If a DBMS lock is available, you can configured it for Oracle databases (see Purge Lock Mechanisms in the Interaction Concentrator User's Guide for the necessary steps). DB2 databases continue to use the previous locking mechanism, which may require manual cleaning of the G_PURGE_STATE table in some rare cases when the purge process has terminated abnormally.

Feature Name	Type of Change	Occurred in Release	Details
			Important If you are running an Oracle database and are planning to use the native purge locking mechanism, *do not* initialize IDB until after you install the Oracle DBMS_LOCK package.
Connection to Interaction Server application type	Additional value	8.1.502.04	If the `cfg` role is enabled, ICON writes the Interaction Server applications to which it is connected into the GC_APPLICATION table (application TYPE=111). If ICON is configured to write data into the DSS tables, the G_DSS_*_PROVIDER tables will have the DBID of the Interaction Server(s) to which it is connected. If you use a modified T-Server application type to connect to Interaction Server (required when using releases prior to 8.1.502.04), this table recorded the T-Server DBID(s) instead.
Support for single-step conferences	Added states and values	8.1.500.04	A new state has been added in the G_PARTY_HISTORY table for the party who initiated the conference with CCEVENT = 12 (Conferenced) and CCEVENTCAUSE = 8 (Single-step Conference). The GSYS_EXT_VCH2 field in the table record contains the DN of the party that was added to the conference. The cause value has been changed in the G_PARTY_HISTORY table for the party added to the conference. Starting with this release, the field contains CCEVENTCAUSE = 2 (Conference) for states with CCEVENT = 2 (Delivered) and CCEVENT = 3 (Established). If the party who initiated the conference drops before the added party answers, the CCEVENTCAUSE field may

Feature Name	Type of Change	Occurred in Release	Details
			have the value 6 (Normal) for the state with CCEVENT = 3 (Established). Single-step conferences are included into total conference count in the G_CALL_STAT table.
Recognition of IWS chat conferences	Added value	8.1.500.04	To provide this support, ICON records a value for the eServices `attr_visibility_mode` interaction attribute in the keyname field of the G_USERDATA_HISTORY table. The attribute value indicates the mode of an agent who has joined a (chat) conference: 1 (Conference mode), 2 (Monitor mode), or 3 (Coach mode, also known as chat consult).

About Interaction Concentrator

This page highlights major ICON features and functionality.

> **Tip**
> For detailed information about many of the features and functions below, see the *Interaction Concentrator User's Guide*.

- For both voice and multimedia, including Genesys eServices (chat and e-mail) and 3rd Party Media, captures and stores information about:
  - The current contact center configuration (objects and associations), and preserves information about deleted configuration objects and terminated associations.
  - Active and completed voice interactions, including switch, DN, time, and routing information about calls and parties. Interaction Concentrator uses a globally unique call identifier.
  - Detailed information about multimedia interactions (e-mail, non-SIP chat, and custom-designed media such as fax and web forms).
  - SIP chat interactions.
  - Agent states and login sessions, for agents handling voice as well as multimedia (e-mail, non-SIP chat, other third-party media) and SIP chat interactions.
  - Detailed information about virtual queue usage in interaction processing, including reporting on a wide range of detailed routing results for interactions that are distributed from virtual queues.
  - Detailed information about interactions that are generated in a network-based contact solution.
  - Detailed information about interactions that are generated in a network call parking environment.
  - Attached data and captures the history of attached data changes.
- For voice interactions:
  - The option to report after-call work (ACW) for the first interaction associated with ACW, as well as the option to suppress the interruption of the ACW and NotReady agent states by interactions coming to, or produced by, the agent.
  - Custom agent states.
  - Detailed information about virtual routing point (VRP) usage in call processing.
  - Customized attached data processing.
- Captures and stores detailed information about outbound campaigns, including:
  - History of campaign processing.
  - History of chain processing.
  - Precalculated metrics provided by OCS.
- Provides a configurable filtering mechanism for certain types of data, to enable the optimization of

database size and performance.

- Provides the ability to resynchronize the configuration data in IDB with Configuration Database on demand.
- Supports high availability (HA) of all types of data through the use of parallel ICON instances, each with its own instance of IDB, in combination with supplementary data that provides information about the availability and reliability of the data stored in IDB.
- Supports near–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 and produce complete data for reporting across sites. Interaction Concentrator provides a stored procedure to merge the interaction records for multi-site interactions.
- Supports multibyte character encoding.
- Stores time information in both Greenwich Mean Time (GMT), as a datetime data type, and Coordinated Universal Time (UTC) seconds, as an integer data type.
  - Obtains the time information from the timestamps of the data provider events (for example, T-Server TEvents), in the form of UTC seconds.
- Provides mechanisms to purge voice and multimedia interactions, agent login session data, attached data, and OCS data that is stored in IDB.
- Supports connections secured by means of Transport Layer Security (TLS) and TLS Federal Information Processing Standards (FIPS) between Interaction Concentrator and the servers to which it connects.
- For enhanced data security, supports encrypted RDBMSs and hiding of TEvent attached data in logs. For more information on these features, see the *Genesys Security Deployment Guide*.

The Interaction Concentrator Components and What They Do

Interaction Concentrator consists of a server, known as the ICON Server or simply ICON, and the Interaction Database (IDB).

ICON

This section provides a high-level overview of how ICON handles data.

Data Preprocessing

ICON preprocesses events received from Configuration Server, T-Server, Interaction Server, and Outbound Contact Server (OCS), according to the role configured for the ICON instance. Preprocessing 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. You can also configure the total number of keep-in-memory interactions that can reside concurrently in an interaction queue or interaction workbin. (This functionality requires Interaction Server release 7.6.1 or higher.)

For more information about the relevant configurations, refer to the following options:

- acc-proc-tout
- acc-queue-lifespan
- acc-queue-size

Data Preparation and Writing

Once data is processed in the in-memory queue, ICON performs the following functions:

- Writes the data from the in-memory queue to the persistent queue and persistent caches.
- Manages the data in the persistent queue and persistent caches.
- Writes data from the persistent queue into IDB.
- Writes data from the persistent cache for configuration data (**cfg-sync.db**) into IDB.

The Persistent Queue (PQ) File

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 requests to write data to IDB. Data in the persistent queue survives a shutdown and restart of ICON. The size of the persistent queue is not formally limited by ICON, but the operating system may impose some limitations.

> **Important**
>
> To reduce the possibility that Interaction Concentrator might lose connection with the **pq** file you are required to locate it on a local drive rather than a network or removable drive.

Each ICON instance creates its own persistent queue file (default name **icon.pq**), which stores data for all the roles that are configured for that ICON. For more information, see ICON Roles.

Persistent Queue Configuration Options

ICON configuration options enable you to specify:

- The file name of the persistent queue.
- The frequency (in terms of number of committed transactions) with which ICON clears data out of the persistent queue.
- Thresholds for environment failure alarms.
- The alarm thresholds can be used to monitor ICON performance.
- Persistent queue behavior at startup.

For more information, see the descriptions of the following options, which control persistent queue behavior:

- agent-pstorage-name
- cfg-dbname
- pq-backlog-alarm-threshold
- pq-backlog-clearance-threshold
- pq-dbname
- pq-purge-number
- pq-startup-check
- pq-startup-purge

Persistent Caches for Configuration and Login Session Data

In addition to the regular persistent queue, the ICON instance that performs the **cfg** role creates and maintains a persistent cache for configuration data. The name of the persistent cache for configuration data is **cfg-sync.db** and it cannot be changed.

The **cfg-sync.db** persistent cache plays an important role in maintaining IDB synchronization with the Configuration Database. ICON keeps a timestamp in the persistent cache for configuration data

changes and, on startup, requests from Configuration Server all configuration changes that occurred after that timestamp.

- For more information about how the persistent queue and the **cfg-sync.db** persistent cache work to maintain up-to-date configuration information, see Populating Configuration Data.
- For recommendations about best practices regarding synchronization, see Resynchronization.

Persistent Cache for Agent Login Session Data

In addition to the regular persistent queue, the ICON instances that perform the **gcc**, **gls**, and **gud** roles create and maintain a persistent cache for agent login session data. In High Availability (HA) deployments, ICON uses this cache to prevent duplicate storage of agent login sessions in IDB and to prevent stuck login sessions. For more information, see Agent States and Login Sessions.

A configuration option, agent-pstorage-name, enables you to specify the name of this persistent cache. The default file name is **apstorage.db**.

IDB

The Interaction Database (IDB) stores data about contact center interactions and resources at a granular level of detail. IDB is a database optimized for storage (in other words, primarily 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 Introducing IDB Schema.
- For a complete table structure and descriptions of all IDB tables and fields, see the *Interaction Concentrator 8.1 Physical Data Model* document for your particular relational database management system (RDBMS).

Stored Procedures

Interaction Concentrator uses a number of stored procedures. Most of these are entirely internal to Interaction Concentrator functioning. Therefore detailed information about them is not relevant for end users.

Most stored procedure names start with a schema-specific prefix, so that they constitute a schema-specific package. Each ICON 8.1.x version works only with the stored procedures package for the associated schema. This streamlines future migration by reducing the number and combinations of scripts that must be executed to upgrade the required stored procedures. A wrapper script links the stored procedures that are exposed for end-user use to the equivalent stored procedures in each schema-specific set.

- For more information about these stored procedures, refer to Using Special Stored Procedures.

The following stored procedures are exposed for end-user use and require user input or action:

Purge Procedures

There are a number of alternative procedures for purging IDB. For a detailed discussion of purge procedures, see Purge Stored Procedures in the *Interaction Concentrator User's Guide*.

- **gsysPurge80/gsysPurge81**—Safely purges voice and multimedia interactions; attached data; agent login session data; and Outbound Contact data from IDB. The version of this purge procedure corresponds with the Interaction Concentrator release you are using.
  - The gsysPurge81 purge procedure has been renamed to GSYSPurge81Common; however, the wrapper name, GSYSPurge81, remains the same as in previous releases, so you do not need to change scripts as a result of this update.
- **purgePartitions811**—In a partitioned IDB, this purge procedure clears the database by truncating all except the specified number of days/partitions in all affected tables. It also retains a default additional partition for "tomorrow."
  - The purgePartitions811 procedure is supported only for IDBs running Oracle 11 or higher.
  - Genesys recommends that you do not use the purgePartitions811 purge procedure on IDBs containing long-living data, such as multimedia IDBs.
- **gsysPurgeIR, gsysPurgeUDH, gsysPurgeLS, and gsysPurgeOS**—Safely purge voice interactions, user data history, agent login session, and Outbound Contact data, respectively, from IDB.
  - These separate purge procedures were discontinued in release 8.1.503.03. If you are using Interaction Concentrator 8.1.503.03 or higher, use gsysPurge81 or (in Oracle environments) purgePartitions811.

> **Important**
>
> These procedures purge all tables that accumulate eventually-obsolete data. However, they do not purge all tables. For a list of tables that are purged, see Tables Purged by the Purge Stored Procedures.

Custom Dispatchers

The following stored procedures customize attached data processing.

- gudCustDISP1
- gudCustDISP2

Merge Procedures

> **Important**
>
> Genesys Info Mart 8.x performs its own merge procedure and does not use the Interaction Concentrator merge procedures.

The following merge procedures finalize data processing of closed single-site and multi-site interactions:

- gsysIRMerge and gsysIRMerge2

The following procedure resets the merge procedure to recover from a failed state:

- gsysIRMergeReset

Time-Setting

The following stored procedure populates the G_TIMECODE table to enable time–interval reporting:

- gsysInitTimeCode

Supported Deployment Scenarios

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 so that they 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 one instance of ICON for each instance of T-Server). 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.

Do you need separate IDBs for voice and multimedia data?

The downstream reporting application might affect your choice of deployment architecture.

- Genesys Info Mart releases earlier than 8.5.007: In deployments that include both voice and multimedia interactions, you must use separate ICON applications to process each type of data and store voice and multimedia data in separate IDBs.
- Info Mart releases 8.5.007 and higher: You are not required to use separate ICON and IDB instances. However, Genesys continues to recommend that you use separate ICON applications and IDBs. Given the different durations for typical voice and multimedia interactions, having separate ICONs and IDBs enables you to tailor the decisions you make about application management, data retention policies, data recovery processes, and so on.

> **Important**
>
> Interaction Concentrator does not support deployments in which two ICON instances are configured for the same role, connect to the same T-Server or set of T-Servers, and write data to the same IDB. For more information about the ICON roles and the rules governing role assignments, see ICON Roles.

Basic Architecture
Single-Site Deployments
Multi-Site Deployments
Network Deployments
Multimedia Deployments

[+] Diagram conventions used in the deployment scenario diagrams

To simplify the deployment diagrams in this section, please note the following conventions:

- DB Server, which enables a connection between ICON and IDB, is omitted from the diagrams for the various deployment scenarios, even though it is required in the overall architecture (as shown in Basic Architecture).
- 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

Integrating with Outbound Contact in the *Interaction Concentrator User's Guide*.

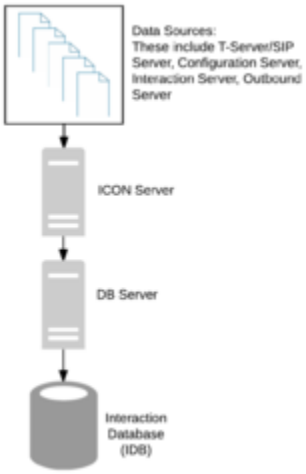
- Interaction Server is not shown for deployments that include eServices. Notes in the text indicate the deployment scenarios that are suitable for multimedia environments. In these environments, the Interaction Server occupies the same position in the architecture as a T-Server.

Basic Architecture

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 applications; it then stores this data into IDB through Genesys DB Server.

The ICON server interfaces with:

- T-Server or Interaction Server to collect data. For more information on this topic, see How ICON Works.
- Solution Control Interface (SCI) or Genesys Administrator, via 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.)
- Message Server, to log messages to the Central Logger.



The figure above depicts the basic ICON architecture, omitting most of the Framework components for the sake of simplicity.

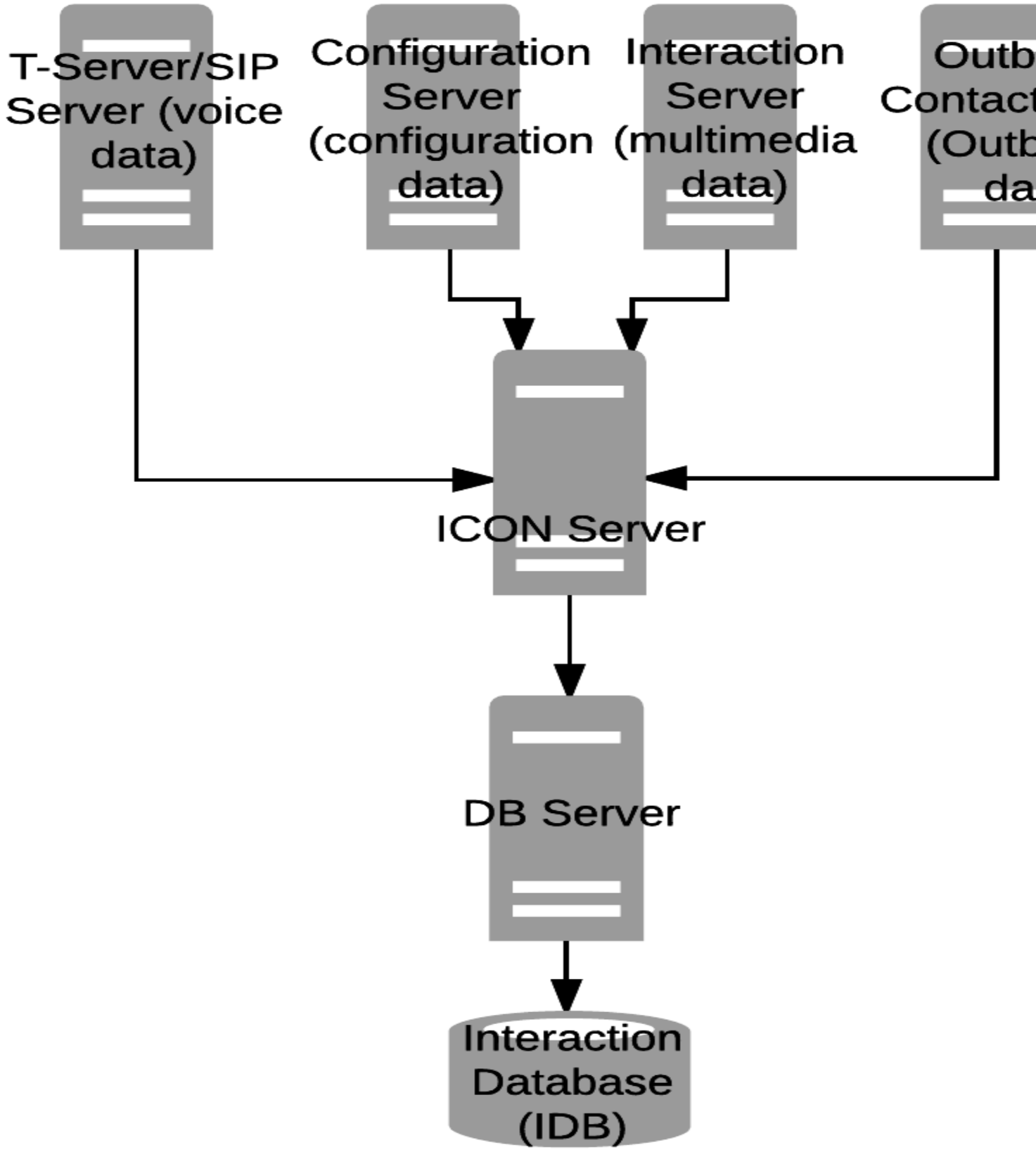
Single-Site Deployments

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

- A single ICON and a single IDB
- Multiple instances of ICON with different roles writing to a single IDB
- Multiple instances of ICON with different roles writing to multiple instances of IDB

One ICON and 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. Deployments with multiple instances of ICON and multiple instances of IDB are straightforward extensions of this model.


T-Server/SIP Server (voice data)
Configuration Server (configuration data)
Interaction Server (multimedia data)
Outb
Contact
(Outb
da
ICON Server
DB Server
Interaction Database (IDB)

One ICON and One IDB

The figure above illustrates the deployment for voice interactions. This type of deployment is also suitable for multimedia environments, in which case the Interaction Server occupies the same position in the architecture as T-Server.

Multi-Site Deployments

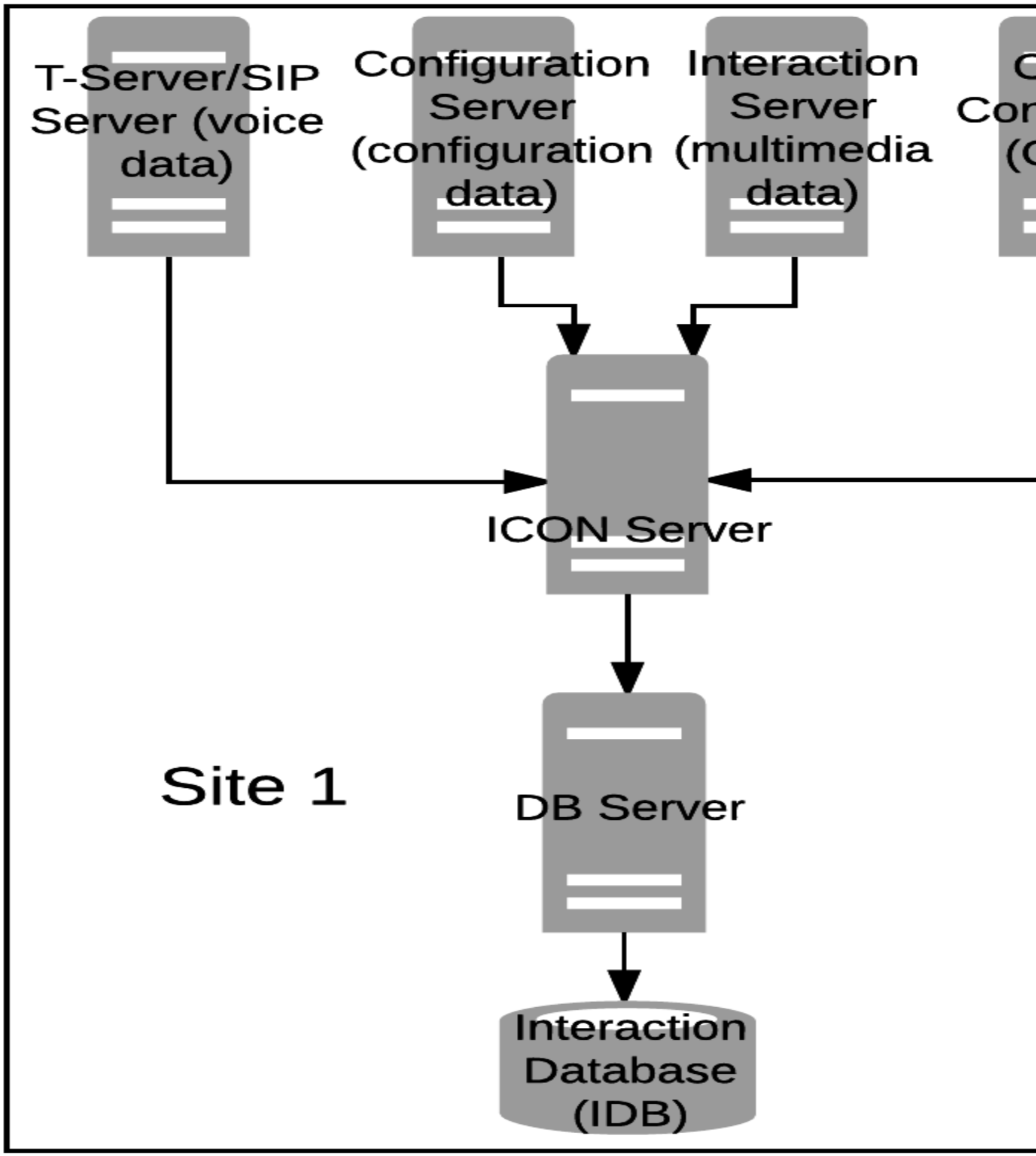
In a multi-site contact center, approaches to Interaction Concentrator deployment vary, depending on network delays between sites, the need for across-the-sites reporting, and other considerations. The following is the basic list of deployments to consider:

- A single ICON instance and a single IDB instance per site
- A single ICON instance and a single, centralized IDB for the entire contact center
- Multiple ICON instances and a single, centralized IDB for the entire contact center

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 others with CTI-related data from the T-Server that serves that site.

The graphic illustrates the deployment for voice interactions. This type of deployment is also suitable for multimedia environments—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.

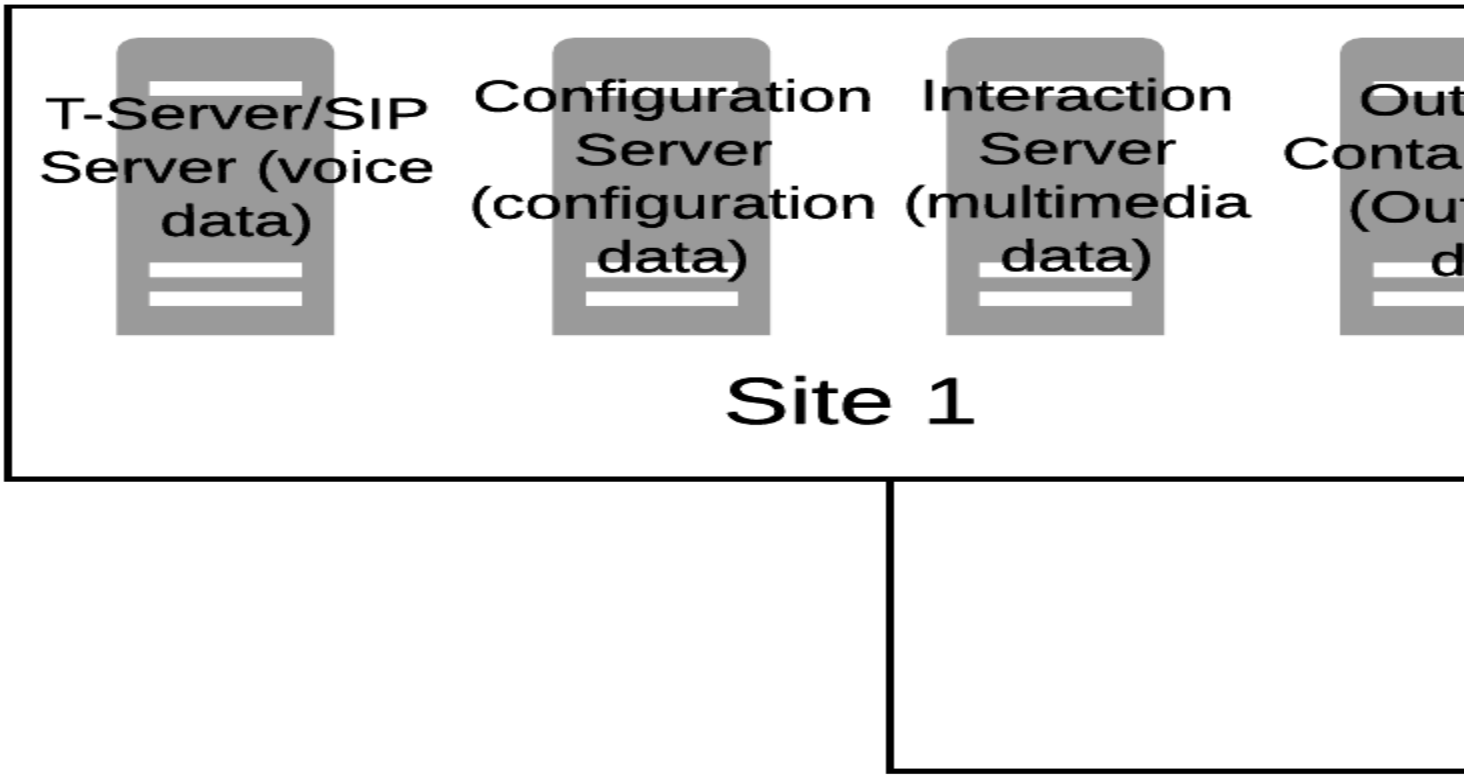

T-Server/SIP Server (voice data)
Configuration Server (configuration data)
Interaction Server (multimedia data)
ICON Server
DB Server
Site 1
Interaction Database (IDB)

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.


T-Server/SIP Server (voice data)
Configuration Server (configuration data)
Interaction Server (multimedia data)
Site 1

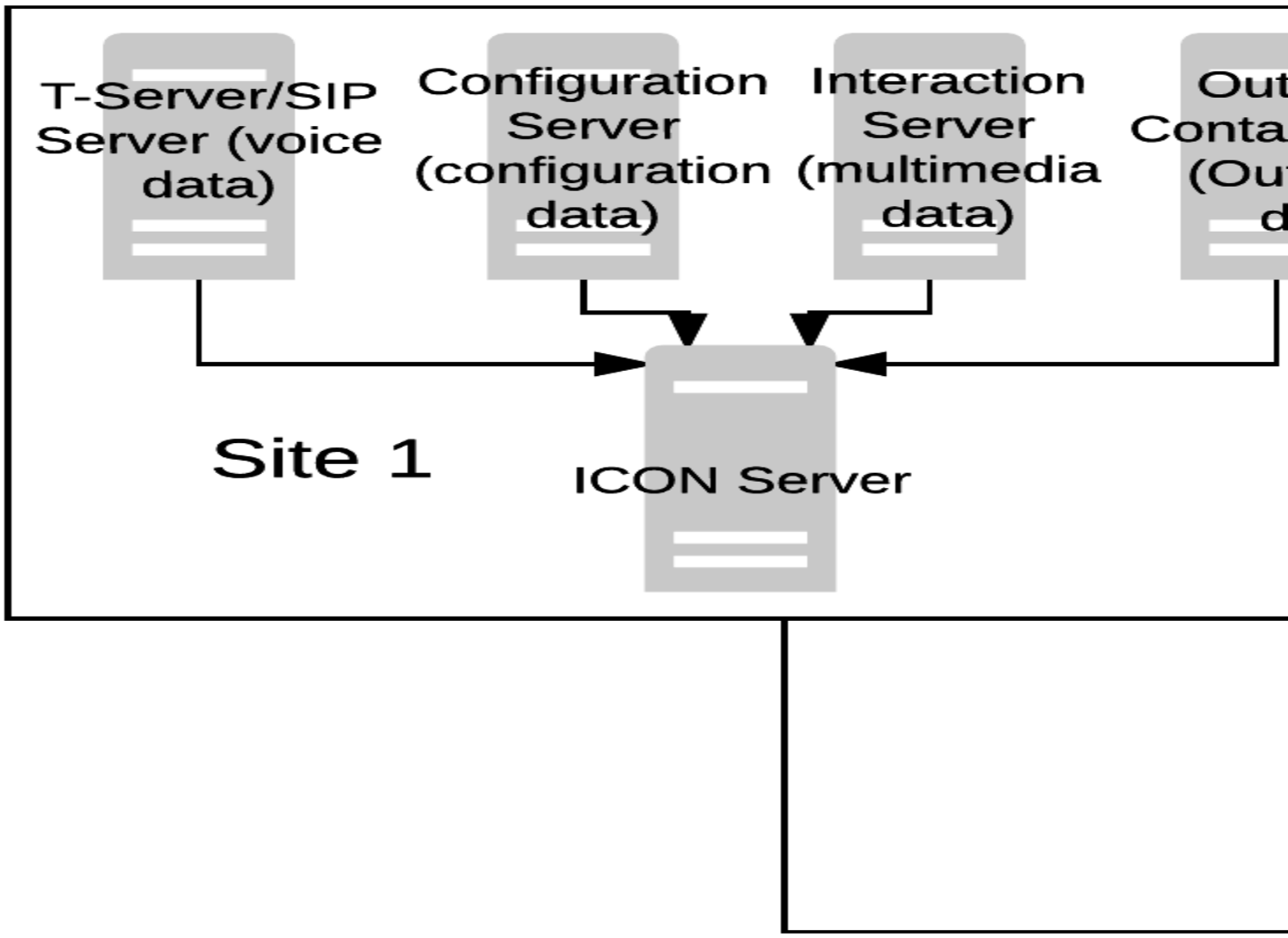
Multi-Site Deployment: A Single ICON and a Centralized IDB Instance

This scenario helps you avoid the need to merge data from different databases. However, note the following considerations:

- If you are running Genesys Info Mart 7.6 or earlier, or running Interaction Concentrator without Genesys Info Mart, you must regularly run the Interaction Concentrator intra-IDB merge stored procedure to ensure correct reporting of multi-site calls (see The Merge Stored Procedure).
- 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, interaction details from all contact center sites come into the same database through separate ICONs.


T-Server/SIP Server (voice data)
Configuration Server (configuration data)
Interaction Server (multimedia data)
Out
Conta
(Ou
d
Site 1
ICON Server

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

> **Warning**
> This functionality is not supported for eServices (email and chat) or 3rd Party Media. Although Interaction Concentrator supports this deployment scenario, Genesys Info Mart does not. Genesys Info Mart requires that each instance of ICON must store data only in an instance of IDB not connected to any other ICON instance.

Like the scenario of one ICON and one IDB for the contact center (see Multi-Site Deployment: A Single ICON and a Centralized IDB Instance), 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, if you are running Genesys Info Mart 7.6 or earlier, or running Interaction Concentrator without Genesys Info Mart, to ensure data correctness for multi-site calls, you must regularly run the Interaction Concentrator intra-IDB merge stored procedure (see The Merge Stored Procedure).

Network Deployments

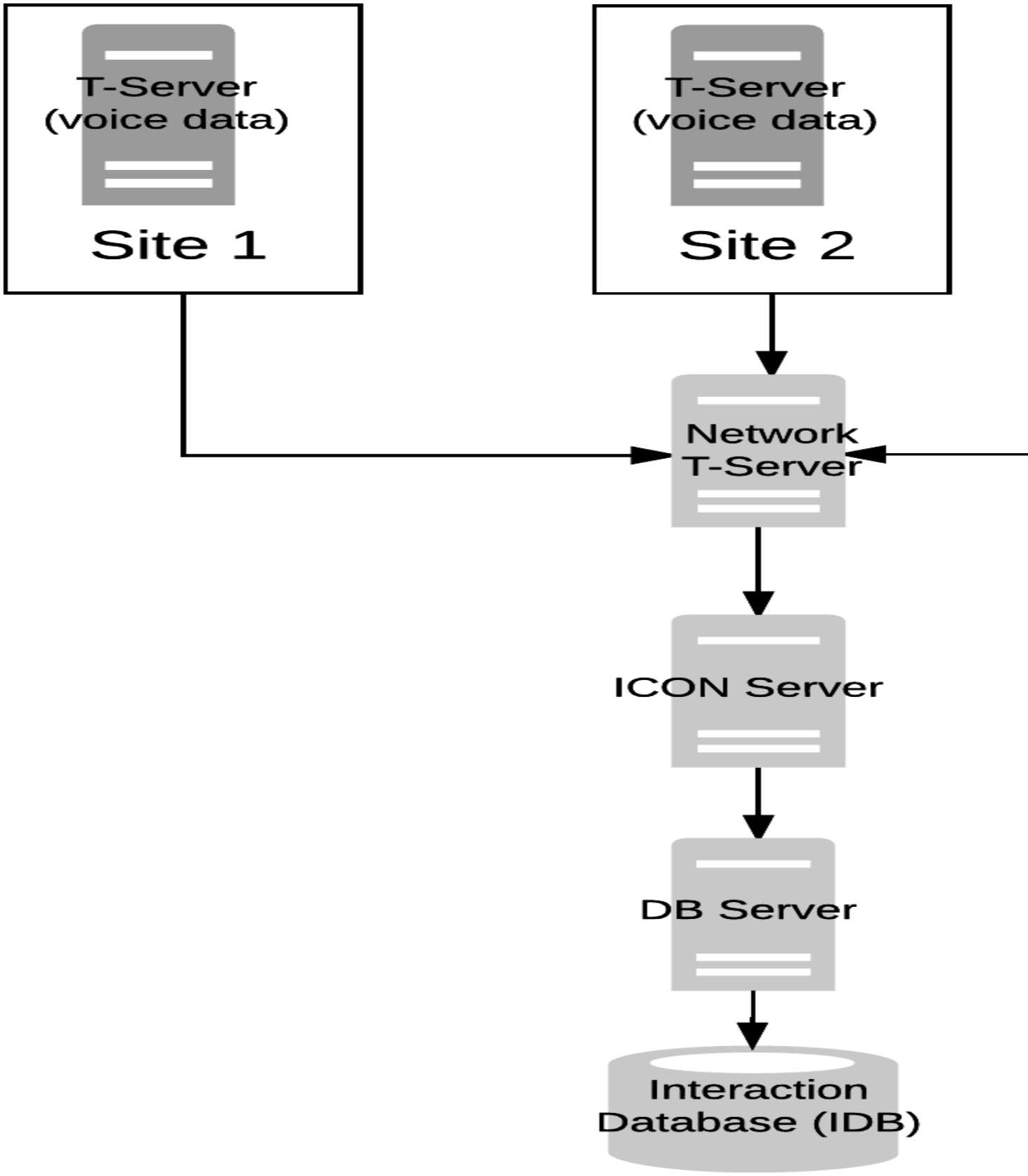
In a network configuration, a number of Premise T-Server applications are connected to a Network T-Server. The ICON instance connects to the Premise T-Server and Network T-Server applications.

Interaction Concentrator supports the following Network T-Server and IVR T-Server deployments:

- A single ICON connected to a single Network T-Server for each network switch
- A single ICON connected to multiple Network T-Servers for each network switch, operating in load-balancing mode
- A single ICON connected to multiple IVR T-Servers for each IVR switch, operating in load-balancing mode

One Network T-Server per Network Switch

This configuration is applicable for both single- and multi-site deployments, in which there are single or multiple Network T-Servers, and each Network T-Server is connected to a separate switch (in other words, the network switch and Network T-Server are not working in load-balancing mode). Each ICON instance can be connected to multiple Network T-Server applications and multiple Premise T-Server applications.

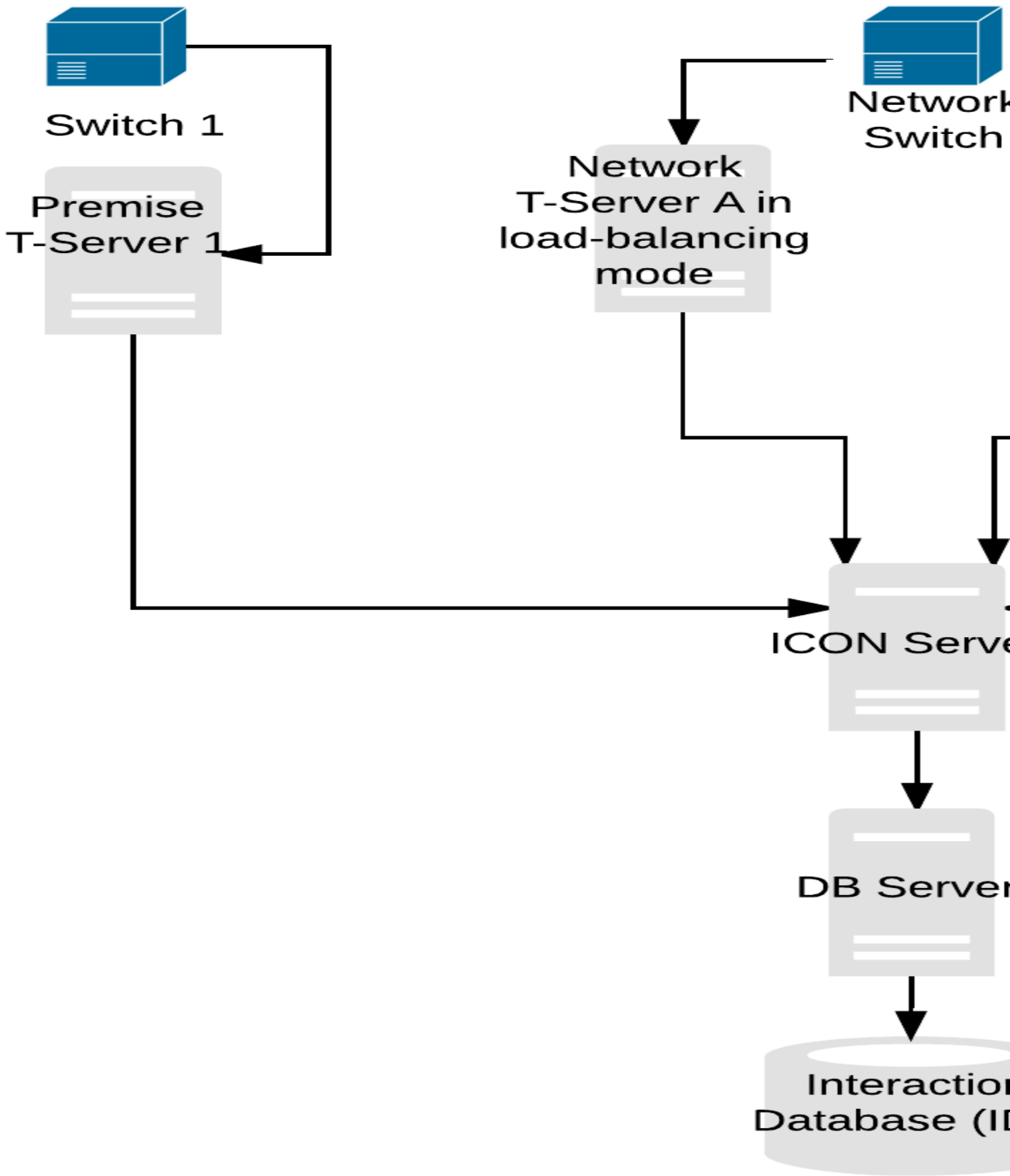

T-Server
(voice data)
Site 1
T-Server
(voice data)
Site 2
Network
T-Server
ICON Server
DB Server
Interaction
Database (IDB)

Network Deployment: A Single Network T-Server per Network Switch

> **Important**
> This Figure does not show the interconnections among T-Server objects that are required for a Network T-Server deployment.

Multiple Network T-Servers per Switch (Load-Balancing Configuration)

If the switch and multiple Network T-Servers have been configured to work in load-balancing mode, ICON supports deployments in which a single ICON instance connects to multiple Network T-Servers that serve the same switch.


Switch 1
Premise
T-Server 1
Network
T-Server A in
load-balancing
mode
Network
Switch
ICON Serve
DB Server
Interactio
Database (I

Multiple T-Servers per ICON-IDB Pair Deployment: Multiple Network T-Servers per Switch (Load-Balancing Configuration)

> **Important**
>
> This Figure does not show the interconnections among T-Server objects that are required for a Network T-Server deployment.

When multiple Network T-Servers serve the same switch in load-balancing mode, the ICON instance creates and maintains a separate connection for each Network T-Server. ICON monitors interaction activity on the switch through the notifications it receives from each Network T-Server.

A configuration option, **switch-multi-links-enabled** (renamed in release 8.1 from **load-balancing-on-network-switch**) , must be set on the Switch object in the Configuration Layer in order for ICON to identify whether the switch and related Network T-Servers are operating in load-balancing mode. For more information about this option, see switch-multi-links-enabled.

Deployments with Network T-Servers in load-balancing mode or with IVR T-Servers in load-balancing mode (described below) are the only configurations in which ICON supports separate connections to multiple T-Servers serving the same switch.

Multiple IVR T-Servers per Switch (Load-Balancing Configuration)

If a switch and multiple IVR T-Servers in an “in-front” configuration have been configured to work in load-balancing mode, ICON supports deployments in which a single ICON instance connects to multiple IVR T-Servers that serve the same switch.

In such a deployment, the ICON instance creates and maintains a separate connection for each IVR T-Server. ICON monitors interaction activity on the switch through the notifications it receives from each IVR T-Server.

A configuration option, **switch-multi-links-enabled**, must be set on the Switch object in the Configuration Layer in order for ICON to identify whether the switch and related IVR T-Servers are operating in load-balancing mode. For more information about this option, see switch-multi-links-enabled.

> **Important**
>
> Deployments with Network T-Servers in load-balancing mode (as described in Multiple Network T-Servers per Switch (Load-Balancing Configuration), above, or with IVR T-Servers in load-balancing mode are the only configurations in which ICON supports separate connections to multiple T-Servers serving the same switch.

Multimedia Deployments

In a multimedia environment ICON connects to one or more Interaction Servers. When ICON connects to multiple Interaction Servers, they each must be associated with a single tenant. The basic functionality works the same way in both single-tenant and multi-tenant environments. The only significant difference is that in single-tenant environments, the tenant name is not specified.

This functionality also works when you are using the T-Server application type to connect to Interaction Server (necessary in all releases of ICON prior to 8.1.502.04).

Tenant-Related Considerations

ICON processes data only from the Interaction Servers associated with tenants included in ICON's Tenants list.

Each Interaction Server should be associated with only one tenant. If ICON finds that an Interaction Server is associated with more than one tenant, it generates a standard-level error message (`20010 Configuration error. Class [errclass] : [errtxt].`) and does not connect to that Interaction Server.

> **Important**
>
> To use an Interaction Server associated with more than one tenant, connect to Interaction Server indirectly via T-Server `Application` objects, as described in Special Connection Procedure for Releases 8.1.500.04 and Earlier, on the Multimedia tab of the Special Configuration Requirements page.

Switch-Related Considerations

Each tenant assigned to an Interaction Server should have only one multimedia switch. Although ICON can complete the connection to Interaction Server if it finds multiple switches of the "Multimedia switch" type assigned to a tenant, this architecture is not considered to be supported and data results might be inconsistent. In this scenario, ICON generates a standard error message (`20010 Configuration error. Class [errclass] : [errtxt].`), chooses the oldest created Switch (the one with the smallest DBID), and proceeds to connect to the Interaction Server(s) associated with that tenant.

If ICON finds that the tenant of an Interaction Server does not have a multimedia type switch, ICON generates a standard-level error message (`20010 Configuration error. Class [errclass] : [errtxt].`) and does not connect to that Interaction Server.

Login Session Consideration

If there are multiple Interaction Servers for one tenant, agents must never have two login sessions from the same place to different Interaction Servers at the same time. If ICON gets a login session from an Interaction Server on a Place when there is another open login session on the same Place, the earlier session will be considered as "stuck."

If you do want to support agents having login sessions to different Interaction Servers at the same time, deploy the Interaction Servers in a cluster. There are two options:

- **Option 1: Use Interaction Server Proxy**
  Connect the ICON(s) to Interaction Server Proxy as described for Interaction Server Cluster in the *eServices Deployment Guide*, particularly the Suggested Deployment Configuration section.
- **Option 2: Only one ICON records agent login session information**
  If your deployment includes Genesys Info Mart release 8.5.016.04 or later, connect the ICON(s) to individual Interaction Servers in the cluster, with at least one ICON (or ICON HA pair) having the `gcc` and `gud` roles and only one ICON (or ICON HA pair) having the `gls` role, and with the Genesys Info Mart populate-sm-busy-from-mm-ixns configuration option set to `true`. For more information, see the Genesys Info Mart 8.5.016.04 release note.

Interaction ID Consideration

If an interaction might be handled by multiple Interaction Servers, the interaction ID must be unique across all Interaction Servers that might handle a single interaction.

Support for Dynamic Changes

ICON supports tracking dynamic changes to Interaction Server properties and connections in the same way as it does for T-Server.

This means that after a host/port change, ICON keeps its connection to the original Interaction Server until there is a disconnection. Interaction Server is required to restart after a host/port change, so ICON will then connect to the new host/port.

ICON will connect to new Interaction Server applications after they are added to ICON.

ICON Roles

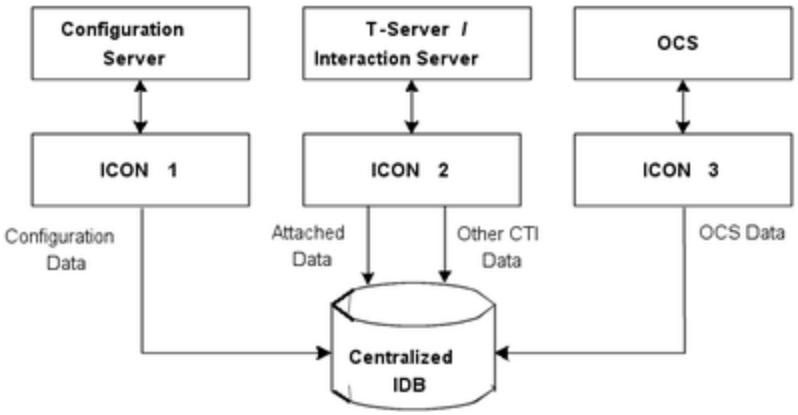
In a contact center that has a large Genesys configuration environment and/or that processes high call volumes, possibly with large amounts of attached data, you can improve Interaction Concentrator performance by deploying multiple ICON instances, each of which collects data only of a certain type.

About Roles

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

- Configuration information—An ICON instance stores the initial contact center configuration state and a history of configuration changes that it retrieves from Configuration Server. ICON stores configuration information about all configuration objects in the environment that are supported by ICON. In a multi-tenant environment, ICON stores configuration information about all of the 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 deployment, ICON stores similar Interaction Server data about multimedia interactions (email, chat, and 3rd Party Media).

  The role that enables ICON to capture interaction-related and party-related information is the gcc role (for more information, see the description of the role option). Regardless of whether you have configured the ICON instance to perform this role, if T-Server or Interaction Server is present on the Connections tab of the ICON Application object, ICON will perform aspects of the gcc role, which is required for internal processing in connection with other roles. However, for an ICON instance to be able to store interaction and party information from T-Server or Interaction Server, it must have the gcc role defined.
- 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, an ICON instance can store OCS data that pertains to outbound calls and campaigns.
- License reporting information—In an environment with License Reporting Manager, you can configure a specific instance of ICON to store license reporting data.



In the example shown in the figure above, 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 or Interaction Server. Finally, the instance named ICON 3 handles OCS data only.

Recommended Role Assignments

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.
- 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.
- If multiple ICON instances are writing data to the same IDB, ensure that you assign only one ICON instance to write configuration data to the IDB. See the Rules and Restrictions tab.
- For LRM, the recommended approach is to use dedicated ICON instances set to the `lrm` role. If the ICON instance must be shared with reporting applications, then ICON should be set to the `gls` and `gos` roles, which can also support LRM. The `lrm` role should *not* be combined with the `cfg` role.

For the values that enable role assignments, see the description of the role configuration option.

The video below explains how to choose the correct role option values for your environment:

Link to video

Rules and Restrictions

When you assign ICON roles, observe the following restrictions:

- Two or more instances of ICON that perform the same role(s) cannot store information from the same data source(s) to the same IDB(s).
  For example, if you have two ICONs, each configured to perform the `gcc`, `gud`, and `gls` roles, they can write to the same IDB only if they are connected to different T-Servers or Interaction Servers.

  Conversely, if you have two ICONs, each configured to perform the `gcc`, `gud`, and `gls` roles, they can be connected to the same set of T-Servers or Interaction Servers only if they write to different IDBs. See Login Session Consideration for an alternative option for Interaction Server Cluster.

- Two or more instances of ICON that perform the `cfg` role cannot store configuration information to the same IDB(s).

- Be aware that the default value of the **role** option is `all`.

- If you have more than one instance of ICON writing to the same IDB, you must configure the ICON applications so that only one ICON performs the `cfg` role.

Role assignments must be configured using only lower case (for example, `cfg`). ICON interprets uppercase (`CFG`) or mixed case (`Cfg`) settings as invalid and defaults to the `all` role.

Planning Your Deployment

This topic provides the following information that you need in order to plan an Interaction Concentrator (ICON) installation:

- Compatibility
- Prerequisites
- Antivirus Guidelines
- General Considerations
- Database Sizing

Compatibility

This section lists the various Genesys components with which Interaction Concentrator release 8.1 is compatible. For information about supported operating systems and relational database management systems (RDBMSs), see the *Genesys Supported Operating Environment Reference Guide*.

The table below lists the Genesys product components with which Interaction Concentrator operates. Refer to *Interaction Concentrator 8.1 Release Note* for any updates to the release requirements for the various components.

Area of Functionality	Component/Product
Configuration Layer	Configuration Server release 7.6 or higher DB Server release 7.6 or higher **Notes:** • Unicode support on a Microsoft SQL IDB requires DB Server 8.1.301.11 or higher. • PostgreSQL IDB requires you to use DB Server release 8.1.301.11 or higher, which uses DB Client 8.5.1. Use the following configuration setting to point DB Server to the 8.5.1 DB Client: `postgre_name=./dbclient_851/dbclient_postgre`. • Configuration Server 8.5 and higher accepts `Person` object Employee ID values of up to 255 characters. However, Interaction Concentrator only supports Employee ID values of 64 characters or fewer. This constraint applies to the value written to the EmployeeID field in the GC_AGENT table.

Area of Functionality	Component/Product
Management Layer	Message Server release 7.6 or higher Local Control Agent release 7.6 or higher
T-Server	T-Server release 7.6 or higher **Note:** The feature to determine which party released the call requires T-Server release 8.0 or higher, and is supported for the Alcatel A4400/OXE switch and Avaya switches (requires Avaya Communication Manager 8.0.101.05 or higher).
eServices (formerly Multimedia or Multi-Channel Routing (MCR))	eServices Interaction Server release 7.5 or higher **Note:** To configure the total number of keep-in-memory interactions that can reside concurrently in an interaction queue or interaction workbin requires Interaction Server release 7.6.1 or higher.
Outbound Contact	Outbound Contact Server release 7.6 or higher
Universal Routing	Universal Routing Server release 7.6 or higher

Prerequisites

Interaction Concentrator has important specific requirements noted in the following subsections. Before you install Interaction Concentrator, review the requirements and recommendations in thefollowing sections:

- Hosting
- Genesys Management Framework
- Telephony Objects
- T-Server
- Outbound Contact
- Universal Routing
- Multimedia
- Interaction Database (IDB)
- Security Features

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 Network Locations for Framework Components 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).
- For better data reliability install ICON as close to T-Server as possible. Ideally, ICON should be on same physical host as the T-Server or Interaction Server for which it gathers data. This prevents network disruptions within the LAN from making an impact on data quality. If you are running a high availability environment, the second instance of ICON should be on a different host but in same subnet as the T-Server or Interaction 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.

For performance reasons, Genesys recommends that you set up the DB Server on the same host 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.

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 or Genesys Administrator/ Genesys Administrator Extension.

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.

> **Important**
>
> For information about, and deployment instructions for, these Framework components, see the *Management 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 and DN Configuration Options.

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.

All T-Servers and, if applicable, Interaction Servers of type T-Server from which an ICON instance should collect data must be listed among the ICON Application object's connections.

> **Important**
>
> 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

To provide outbound information to ICON, at least one OCS application must exist and be properly configured. 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 *Genesys Migration Guide*. For recommendations on how to enable outbound reporting in Interaction Concentrator, refer to Integrating with Outbound Contact.

Universal Routing

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

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

Interaction Concentrator functionality related to writing extended routing results from virtual queues into IDB requires Universal Routing Server (URS) release 7.6 or higher. To enable extraction of this extended routing information, you must also set the values of the **report_reasons** and **report_targets** configuration options in URS to `true`.

For deployment instructions for Universal Routing components, see the *Universal Routing 8.1 Deployment Guide*. For Universal Routing migration instructions, see the *Genesys Migration Guide*. For recommendations on how to enable virtual queue reporting in Interaction Concentrator, refer to Monitoring Virtual Queues and Routing Points.

Interaction Concentrator functionality related to storing virtual queue history in IDB requires URS release 8.1 or higher.

Multimedia

If you intend to collect interaction, agent state, and agent login session data for eServices (email and chat) and/or 3rd Party Media interactions, configure and install Interaction Server, if it is not yet deployed (for instructions, see the *eServices Deployment Guide*).

- For deployment architectures supported by ICON, see the Multimedia tab of the Supported Deployment Scenarios page.

The Interaction Server from which an ICON instance should collect data must be listed among the connections of the ICON Application object. (Or, if you are running a release of ICON prior to 8.1.502.04, the connection to Interaction Server will be represented by a T-Server object. For more information, see Configuring for Multimedia Data.)

> **Important**
>
> The functionality introduced in ICON release 7.6.1 to support a large number of concurrently active multimedia interactions requires Interaction Server release 7.6.1 or higher.

For information about—and deployment instructions for—Interaction Server, see the *eServices*

Deployment Guide. For recommendations on how to enable multimedia reporting in Interaction Concentrator, refer to Integrating with eServices and 3rd Party Media in the *Interaction Concentrator 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. For full information about supported RDBMSs and RDBMS versions, see the *Genesys Supported Operating Environment Reference Guide*. For the changes in RDBMS support introduced with Interaction Concentrator 8.1, see the *Interaction Concentrator 8.1 Release Note*.

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

- Review the information about the IDB structure in Introducing IDB Schema.
- Estimate IDB size, using the *Interaction Concentrator 8.1 Database Size Estimator*. For more information, see the Database Sizing tab on this page.
- To improve performance, locate your ICON IDB and the associated DB Server application close together in your network topology.

Security Features

Interaction Concentrator supports the following security features:

- Encrypted RDBMSs
- Hiding TEvent attached data from logs.
- Hiding sensitive attached data information in the ICON log file at all log levels.

Important

- If any attached data is configured to be hidden, ICON debug-level messages may hide all attached data keys and values, not just the keys configured to be hidden.
- The **default-filter-type** option in the **[log-filter]** section and the **<key-name>** options in the **[log-filter-data]** section are configured in the ICON Application object. However, the description of the default and valid values, and a full explanation of how these options work, are located in Hide Selected Data in Logs (in the *Genesys Security Deployment Guide*).

Support for Secure Connections

- Starting with release 8.1.1, Interaction Concentrator supports Transport Layer Security (TLS) and TLS-FIPS connections.
- Starting with release 8.1.2, Interaction Concentrator Supports client-side port definition, to provide

secured connections to T-Server, SIP Server, Configuration Server, and Message Server.

On Windows platforms, support for TLS is integrated into the operating system, and there are no additional requirements to enable Interaction Concentrator to support it. On UNIX-based platforms, you must install the Genesys Security Pack on the Interaction Concentrator host.

For details on all of the supported security features, see Security Features in the *Interaction Concentrator User's Guide* and the *Genesys Security Deployment Guide*.

Antivirus Guidelines

Antivirus software can affect system performance and response time but may be necessary to prevent and detect viruses. Genesys recommends that you keep antivirus software enabled on the host where the ICON application is running and monitor performance to ensure there is no impairment to ICON, or to other applications running on the same host.

If you find that antivirus scanning has a significant performance impact, consider excluding the following files and folders from antivirus scanning.

- The **log** folder.
- The three files ICON creates to store information before writing it to IDB. The filenames are configured in the cfg-dbname, pq-dbname, and agent-pstorage-name options.

To reduce the risk of disabling antivirus scanning, Genesys recommends that you do not put any executable files into unscanned folders and that you prohibit **execute** permissions on those folders.

> ### Important
> The recommendation to enable antivirus scanning assumes that your antivirus software is configured to enable all port connections and communication Interaction Concentrator requires. If any required port access or communication is blocked, ICON operations will be affected.

General Considerations

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, and answer the following questions:

- How many ICON servers do you need, and what data should each ICON instance handle? From what sources will the data come to a given ICON instance?
- How many Interaction Databases do you need, and what data will each IDB instance store? Which ICON

instances will store the data into a particular IDB instance? In the case of multiple IDB instances, will you need to deploy a centralized IDB, and, if so, from which subset of IDBs will data be merged into the centralized IDB? How often should the merge procedure be run?

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

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

Database Sizing

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 8.1 Database Size Estimator*, is a Microsoft Excel spreadsheet.

For more information about database sizing and deployment guidelines, see the Interaction Concentrator section in the *Genesys Hardware Sizing Guide*. The Genesys Info Mart section of the Genesys Hardware Sizing Guide can also provide helpful sizing and performance information for Interaction Concentrator.

Configuration and Installation

This page provides instructions for configuring and installing Interaction Concentrator, the Interaction Database (IDB), DB Server, and a Database Access Point (DAP). (DB Server and the DAP are Framework components required to use IDB.)

The following pages contain additional information important for a successful deployment:

- Special Configuration Requirements
- Configuration Options

> **Important**
> The information on this page 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 Appendix: Migration Procedures before you perform any installation procedures.

This section provides instructions on the following topics:

- Deployment Order
- DB Server Deployment
- ICON Deployment
- IDB Deployment
- DAP Deployment

Deployment Order

Before you deploy Interaction Concentrator, review Planning Your Deployment, 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 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 the *Interaction Concentrator Options Reference* sections on Switch Configuration Options and DN Configuration Options.

3. DB Server

> **Tip**
> If you decide to use a DB Server that serves another application for Interaction Concentrator storage purposes, you can 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 *Management Framework Deployment Guide*.

See DB Server Deployment for notes specific to installation with Interaction Concentrator.

4. Interaction Concentrator

Configure and install Interaction Concentrator.

In an environment with multiple ICON instances, repeat the configuration and installation steps for each ICON instance, making the necessary adjustments when you configure ICON connections and configuration options.

5. Interaction Database

- Install and initialize a database for ICON data on one of the supported RDBMSs.

In an environment with multiple IDB instances, repeat the steps to create IDB for each IDB instance.

6. Database Access Point

1. Deploy a DAP `Application` object that specifies IDB connection parameters.
2. In an environment with multiple ICON instances, each of which stores data to its own IDB instance, repeat the steps 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.
3. Ensure that the role configuration option settings that you specify for the DAP are consistent with the **role** settings 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:

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 performance reasons, Genesys recommends that you set up the DB Server on the same host 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.

Important

- If you are running an Oracle database, make sure the NLS_LANG environment variable on the host where DB Server is installed is set to match the character encoding of data in the Oracle database; for example, `NLS_LANG=.UTF8`. This is especially important when using UTF-8 language encoding, and in all Chinese, Japanese, and Korean language environments. For additional information on specific configuration information for Oracle databases, see the Framework Database Connectivity Reference Guide.
- Unicode support on a Microsoft SQL IDB requires DB Server 8.1.301.11 or higher.
- PostgreSQL IDB requires you to use DB Server release 8.1.301.11 or higher, which uses DB Client 8.5.1. Use the following configuration setting to point DB Server to the 8.5.1 DB Client: `postgre_name=./dbclient_851/dbclient_postgre`.

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

Deploying ICON

This section explains how to deploy as many instances of Interaction Concentrator as you need on either a UNIX or a Windows operating system.

> **Important**
>
> Interaction Concentrator supports silent installation on Windows and Unix platforms. Also known as a silent setup, it enables you to install Interaction Concentrator without having to monitor the setup or provide input via dialog boxes or the command line. For instructions, see Silent Setup in the *Framework 8.1 Deployment Guide*.

Environment Assumptions
The instructions in this section assume that you are creating new `Application` objects under the Environment folder, 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.

To deploy the ICON server, perform the following procedures:

1. Import the Application Template
2. Create an ICON Application Object
3. Configure the General tab
4. Configure the Server Info tab
5. Configure the Start Info tab
6. Configure the Options tab
7. Configure the Tenants tab
8. Configure the Connections tab
9. Perform the Installation

Import 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. You can use this `Application` template to create as many `Application` objects of the same type as you need.

> **Important**
>
> For an explanation of how to use Genesys Administrator Extension to import the `Application` template and to create a new `Application` object, see the Genesys Administrator Extension Help file, which is directly available from Genesys Administrator Extension user interface, as well as from the link given here.

1. Open the Configuration Manager main window in Genesys Administrator Extension.
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 8.1 product CD, and open the **TEMPLATES** folder.
6. Select the template file for Interaction Concentrator; it is called **Interaction_Concentrator_81x.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.

The next step is to configure an Interaction Concentrator `Application` object.

Create 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.

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. On each of the Application tabs, enter the settings appropriate for your environment, as explained in detail in the section that follow.
6. Click **OK** to save your changes, then close the **Properties** dialog box.

Configure the General tab

In the **Properties** dialog box, click the **General** tab, and then enter a name for this application.

> **Tip**
> The application template provides information about the application type and version. Interaction Concentrator uses the **Call Concentrator** application type for its Application object in the Configuration Layer.

Configure the Server Info tab

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 a numeric value for a port that is not used by another application. Valid values are in the range of 1–65,535. ICON uses this value as the default listening port for the web interface connection.

> **Important**
> In IPv6 deployments, you cannot set the IP address of the host—only IPv4 addresses can be set for the host. Therefore, enter the *name* of the host instead.

Configure the Start Info tab

> **Tip**
> The properties you define here are updated automatically during the installation procedure.

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 the Starting and Stopping page.

Configure the Options tab

Click the **Options** tab, and then specify or change the values of the configuration options, as suitable for your deployment.

- For information about specific configuration requirements to enable ICON to capture and store various types of data (such as voice, multimedia, Outbound, and so on), see Special Configuration Requirements.
- For information about the entire set of ICON configuration options, see the *Options Reference*.

Role Option

- If you have already deployed another ICON instance that writes to the same IDB, 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 description on the role configuration option and in the video below. For more information about role assignments and restrictions for the `cfg` role, see "Recommended Role Assignments" on the ICON Roles page.

Link to video

Log Options

- Configure both ICON-specific log options and common log options in the log-related configuration sections. For option descriptions, see the log Section of the *Interaction Concentrator Options Reference* and the *Framework Configuration Options Reference Manual*.

> ### Important
> If the Interaction Concentrator working directory differs from the directory where 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.

Running Multiple ICON Applications from the Same Directory

You can run multiple ICON Applications from the same directory using the same executable. To do so, each ICON `Application` should have a different value for the three following options: pq-dbname, cfg-dbname, and agent-pstorage-name. If the ICON `Applications` are configured to write log data to a file or files, these filenames should also be different.

HTTP Listener

> **Important**
> Support for use of the HTTP Listener to monitor and report on Interaction Concentrator performance has been discontinued in release 8.1.512.08 and higher.

- Configure an HTTP listener by creating a **[listeners]** section and specifying the appropriate option. For option descriptions, see the listeners Section of the *Interaction Concentrator Options Reference*.

Configure the Tenants tab

> **Tip**
> 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. It is important to add *all* tenants from whose resources (switches, DNs, agents, and, if applicable, Outbound Contact objects) 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.

Configure the Connections tab

Click the **Connections** tab, and then add the following connections:

- T-Server
- Interaction Server
- Outbound Contact Server
- Configuration Server
- Message Server
- DAP

> **Important**
> If you make any dynamic changes to the Interaction Server or Outbound Contact Server applications listed on the Interaction Concentrator `Application` **Connections** tab, restart ICON to ensure that the changes take effect.

ADDP

For any connections between the ICON instance and its data sources, you can configure the connection to use Advanced Disconnect Detection Protocol (ADDP). To enable ADDP for a connection, specify addp as the Connection Protocol when you configure the connection between the Applications, and set the values for the **Local Timeout**, **Remote Timeout**, and **Trace Mode** properties. For more information, see see Advanced Disconnect Detection Protocol in the *Management Framework Deployment Guide*.

T-Server

If this ICON instance is configured to process CTI-related data, add a connection to T-Server. Note the following special requirements:

- If you have a simple multi-site topology that includes one ICON instance and multiple T-Servers, add a connection to each T-Server. Each T-Server `Application` object must have a `Switch` object assigned to it.
- In a deployment with Network T-Server, add a connection to the Network T-Server. The Network T-Server `Application` object must have a network `Switch` object assigned to it.
- In a deployment in which the Network T-Servers function in load-balancing mode, add a connection to each Network T-Server. For more information, see "Multiple Network T-Servers per Switch (Load-Balancing Configuration)" on the Supported Deployment Scenarios page.

> **Important**
> Starting in release 8.1.4, if host or port information changes for any T-Server listed on the **Connections** tab, Interaction Concentrator dynamically reconnects using the new connection parameters. While ICON disconnects from the prior host/port and connects to the new one, there might be a brief gap in data received from the T-Server.
> In releases 8.1.0 and 8.1.1, you must restart ICON for updates to listed T-Servers to take effect.

Interaction Server

If this ICON instance is configured to process multimedia interactions reported by Interaction Server, add a connection to Interaction Server.

- Restart ICON after adding or removing Interaction Server applications from the ICON **Connections** tab.

> **Important**
>
> If you are installing ICON in a deployment that will use Genesys Info Mart to report on both voice and multimedia interactions, Genesys strongly recommends that you create separate ICON instances and separate IDBs for the voice and multimedia data. See Do you need separate IDBs for voice and multimedia data? for more information.

Outbound Contact Server

If this ICON instance is configured to process outbound data in an environment with Genesys Outbound Contact, add a connection to one or more Outbound Contact Server `Applications`. See Integrating with Outbound Contact in the *Interaction Concentrator User's Guide* for in-depth configuration information.

- Restart ICON after adding or removing Outbound Contact Server applications from the ICON **Connections** tab.

Configuration Server

Add a connection to the Configuration Server application (named **confserv**). You can enable ADDP for connections to Configuration Server in the same way as for other connections.

Genesys recommends that, at a minimum, you add a connection to Configuration Server in the ICON instance that is configured to process configuration data (`cfg` role). However, to minimize the number of unidentified call segments that might occur as a result of missing configuration data, Genesys recommends that you configure an ADDP connection to Configuration Server in each ICON `Application` in your deployment, regardless of the role of the ICON `Application`.

Message Server

If you installed the Management Layer, add a connection to Message Server, in order to provide alarm signaling and centralized logging capabilities. 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 Configuration Manager Help*.

DAP

After you configure one or more DAP `Application` objects, add any DAP `Application` objects through which this ICON instance will access IDBs to the ICON **Connections** tab.

High Availability

In high availability (HA) environments using primary and backup pairs of servers, the servers listed on the **Connections** tab are handled as *primary*. To specify the backup servers for any primary servers,

open the `Application` object for the primary server and add the backup server on the primary server's **Server Info** tab.

Perform the Installation

You can install ICON on either a Windows or a Unix-based system.

Prerequisites

- You have created and configured an Interaction Concentrator `Application` object in the interface you use for configuration, as described above.

Installing on Windows

To install, perform the following steps:

1. Insert the Interaction Concentrator CD into the CD-ROM drive of the machine on which you want to install Interaction Concentrator; or, download the Interaction Concentrator IP to the desired location on the target machine.
2. Navigate to, and open, the **.../windows** directory.
3. Double-click the **setup.exe** file, and then follow the directions in the installation wizard.

Installing on UNIX

To install, perform the following steps:

1. Insert the Interaction Concentrator CD into the CD-ROM drive of the machine on which you want to install Interaction Concentrator; or, download the Interaction Concentrator IP to the desired location on the target machine.
2. Locate the correct installation directory for your platform — for example, **.../solaris**.
3. Save the contents of this directory to a local folder.
4. Locate and run the **install.sh** shell script. Enter requested information when you are prompted to do so.

Deploying IDB

To create a new Interaction Database (IDB), ask your Database Administrator to create a new database for each IDB instance that you intend to deploy for ICON data storage. You can use any of the supported RDBMSs to host your IDB. Then, initialize each IDB instance, using the appropriate scripts for your environment and following the initialization procedures provided in this topic.

> ### Important
>
> - If you plan to use Genesys Info Mart for your downstream reporting, ensure that the connection-related settings for the RDBMS server(s) hosting IDB are suitable for Genesys Info Mart. If your IDBs and Info Mart database are not hosted by the same RDBMS instance, ensure that you set connection-related parameters for your IDB RDBMS server(s) as described on Database Tuning in the *Genesys Info Mart Deployment Guide*.
> - The user account that is created for IDB must have permissions to create database objects such as tables, stored procedures, and sequences.
> - If you are running an Oracle database and are planning to use the native purge locking mechanism, *do not* initialize IDB until after you install the Oracle DBMS_LOCK package.
> - If you need to reconfigure Interaction Concentrator to work with a different IDB instance, or you re-initialize your IDB, see Changing IDB Instances or Re-Initializing Your Current IDB for instructions.

Internationalization Settings

- If you require UTF-8 support, you must use Configuration Layer components of release 8.1.3 or higher.
- If you require Unicode support on a Microsoft SQL database, you must use DB Server 8.1.301.11 or higher. To use Unicode on a Microsoft SQL database, run the **CoreSchema_multilang_mssql.sql** database initialization script rather than the **CoreSchema_mssql.sql** initialization script. For additional details on how to configure Interaction Concentrator to store Unicode data, see Configuring for Unicode Support in an Environment with a Microsoft SQL IDB.
- If you are using a Unicode PostgreSQL IDB, be sure to have client encoding set to UTF-8 as well. If you encounter an error message indicating: `invalid byte sequence for encoding`, you can enforce client encoding to UTF-8 by setting the `PGCLIENTENCODING` environment variable to `UTF8` for the DB Server environment.
- If you are running an Oracle database, make sure the NLS_LANG environment variable on the host where DB Server is installed is set to match the character encoding of data in the Oracle database; for example, `NLS_LANG=.UTF8`. This is especially important when using UTF-8 language encoding, and in all Chinese, Japanese, and Korean language environments. For additional information on specific configuration information for Oracle databases, see the Framework Database Connectivity Reference Guide.

- In environments with PostgreSQL RBDMS, the IDB processing configuration details (having the **role** option set to `cfg`) should be created to use encoding identical to that used by Configuration Server. For example, you may set PostgreSQL database encoding to `WIN1252` to use extended ASCII from European languages, or use SQL_ASCII.
- In environments with PostgreSQL RBDMS, if you require ICON to process user data using any symbols other than ASCII, you must create your PostgreSQL IDB with encoding that supports these symbols. If this encoding is different from the Configuration Server encoding, you must have a separate IDB for configuration details.

For additional details, see Configuring Interaction Concentrator for Multi-Language Support.

Sample Script

Genesys provides the **SampleProc_*db_type*.sql** script to help you understand how you can modify the stored procedures for customized attached data processing. Do not execute the sample script during installation. For more information about configuring your ICON application to support customized attached data processing, see Configuring for Attached Data. For an example of a script to create a custom dispatcher stored procedure and custom storage table, see Sample Script for Custom Attached Data.

Initialization Scripts

After you install the ICON application, the **scripts** subfolder in the directory to which you installed ICON contains a set of initialization, migration, and sample scripts for each RDBMS type. See the Table of Initialization Scripts (below) for a list of these scripts and the purpose of each. In the script names, *db_type* is a placeholder for the specific RDBMS type (`db2`, `mssql`, `postgre`, or `ora` [for Oracle]).

Warning

If you are migrating from an existing IDB, do not simply apply all the scripts listed under Initialization Scripts below. To avoid damaging or erasing existing data, follow the migration procedures that are described in Appendix: Migration Procedures.

If you are running Genesys Info Mart 8.1.1 or 8.1.0, see Scripts Required for Environments Running Genesys Info Mart 8.1.1 or 8.1.0 below.

Table of Initialization Scripts

Script Name	Description
CoreSchema_*db_type*.sql	(For initial installation only) Creates the core IDB tables and indexes.

Script Name	Description
	To use Unicode on a Microsoft SQL database, run the **CoreSchema_multilang_mssql.sql** database initialization script rather than the **CoreSchema_mssql.sql** initialization script.
Upgrade_*target_database_version_db_type*.sql	(For migration only) Upgrades the IDB schema. The scripts you must execute depend on the releases from which and to which you are upgrading. For more information, see the instructions in Appendix: Migration Procedures.
CoreProcedures_*db_type*.sql	Creates the database schema–specific set of stored procedures that implement core ICON functionality, including the merge procedures and the separate procedures to purge different types of data. Tip The script creates default (empty) custom dispatchers named **gudCustDISP1** and **gudCustDISP2** without first dropping any existing stored procedures. 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, the script returns an error, which you can safely ignore.
Purge2_*db_type*.sql or (in special scenarios) **Purge2_PartitionType0_ora.sql**	Creates the **gsysPurge80** or **gsysPurge81** stored procedure. The version created corresponds to the release of Interaction Concentrator you are installing. For a non-partitioned Oracle IDB with the **partition-type** configuration option set to 0, your best performance will be achieved with the **Purge2_PartitionType0_ora.sql** script (available in release 8.1.505.05 and higher). All other non-partitioned environments should use **Purge2_*db_type*.sql**.
CoreSchemaPart_ora.sql	(Optional, for Oracle RDBMSs only) Creates tables, sequences, and indexes for use in a partitioned schema.
PurgePart_ora.sql	(Optional, for Oracle RDBMSs only) Creates the **purgePartitions811** stored procedure required to purge a partitioned Oracle IDB.
Wrapper_for_*schema version_db_type*.sql	Links generically named merge and purge procedures to the equivalent, schema-specific stored procedures in the new set.
drop_*schema version_db_type*.sql	(Optional, for migration only) Removes the set of stored procedures for the specified Interaction Concentrator schema version.
SampleProc_*db_type*.sql	Serves as a sample script, illustrating how to create a custom attached data storage table and modify the custom dispatcher stored procedures.

Scripts Required for Environments Running Genesys Info Mart 8.1.1 or 8.1.0

For environments that include Genesys Info Mart 8.1.1 or an earlier 8.1.x release, run the appropriate script as described in this section every time you migrate to a new release of Interaction Concentrator. (Starting with release 8.1.4, Genesys Info Mart automatically runs the scripts when required.)

> **Tip**
>
> For the location of the scripts and detailed instructions, see the *Genesys Info Mart 8.1 Deployment Procedure* or the *Genesys Info Mart 8.1 Deployment Guide*.

- For a Voice details IDB, use **update_idb_for_gim.sql**.
- For a Multimedia details IDB, use **update_idb_for_gim_mm.sql**.
- For a Configuration details or an Outbound Contact details IDB, use either **update_idb_for_gim.sql** or **update_idb_for_gim_mm.sql**.

Initializing IDB

For a first-time initialization of IDB, follow the RDBMS-specific instructions in the following procedures:

Initializing IDB on DB2

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

1. Go to the directory into which you installed ICON.
2. Go to the **scripts\db2** subdirectory.
3. Execute the following scripts in the order shown:
    1. **CoreSchema_db2.sql**
    2. **CoreProcedures_db2.sql**
    3. **Purge2_db2.sql**—This script is optional. Execute this script if you want to use the **gsysPurge80**/**gsysPurge81** stored procedure in your deployment.
    4. **Wrapper_for_*schema version*_db2.sql**—Execute this script if your deployment will use the **gsysIRMerge** or **gsysIRMerge2** merge procedure, or if your deployment will use the **gsysPurgeIR**, **gsysPurgeUDH**, **gsysPurgeLS**, or **gsysPurgeOS** purge procedures.
4. To execute the scripts:
    - Insert the following command line at the beginning of each script, providing appropriate values for the placeholders:

        ```
        db2 connect to dbname user user using password
        ```

- Use the following command line to load each initialization script:

```
db2 -w -td@ -fscript_name
```

Initializing IDB on Microsoft SQL

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

1. Go to the directory into which you installed ICON.
2. Go to the **scripts\mssql** subdirectory.
3. Execute the following scripts in the order shown:
    1. **CoreSchema_mssql.sql** or, for environments requiring Unicode support, **CoreSchema_multilang_mssql.sql**
    2. **CoreProcedures_mssql.sql**
    3. **Purge2_mssql.sql**—This script is optional. Execute this script if you want to use the **gsysPurge80/gsysPurge81** stored procedure in your deployment.
    4. **Wrapper_for_*schema version*_mssql.sql**—Execute this script if your deployment will use the **gsysIRMerge** or **gsysIRMerge2** merge procedure, or if your deployment will use the **gsysPurgeIR**, **gsysPurgeUDH**, **gsysPurgeLS**, or **gsysPurgeOS** purge procedures.
4. To execute the scripts, use the following command line to load each initialization script, providing appropriate values for the placeholders:

```
sqlcmd -S dbms_server -d dbname -U user -P password -i script_name
```

where *sqlcmd* is `isql.exe` or `osql.exe`

Initializing IDB on Oracle

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

1. Go to the directory into which you installed ICON.
2. Go to the **scripts\oracle** subdirectory.
3. Execute the following scripts in the order shown, unless you are creating a partitioned Oracle database, in which case use the list of scripts that follows the standard set.
    - To execute the scripts, log in to the **sqlplus** command processor, and type the following at the command prompt:

```
@ script_name
```

    - For a standard Oracle IDB, run the following scripts:

    1. **CoreSchema_ora.sql**
    2. **CoreProcedures_ora.sql**
    3. **Purge2_ora.sql** or **Purge2_PartitionType0_ora.sql**—These scripts are optional. Execute one of these scripts if you want to use the **gsysPurge80/gsysPurge81** stored procedure in your deployment. **Purge2_PartitionType0_ora.sql** (available in release 8.1.505.05 and higher) is specifically optimized for a non-partitioned Oracle IDB with the **partition-type** configuration option set to `0`. Other environments should use **Purge2_ora.sql**.

4. **Wrapper_for_*schema version*_ora.sql**—Execute this script if your deployment will use the **gsysIRMerge** or **gsysIRMerge2** merge procedure, or if your deployment will use the **gsysPurgeIR**, **gsysPurgeUDH**, **gsysPurgeLS**, or **gsysPurgeOS** purge procedures.

- If you are using partitioning on Oracle, run the following scripts instead:

1. **CoreSchemaPart_ora.sql** (instead of **CoreSchema_ora.sql**)
2. **CoreProcedures_ora.sql**
3. **PurgePart_ora.sql** (instead of **Purge2_ora.sql**).Execute this script if you want to purge a partitioned IDB by truncating partitions. This purge method can speed up the purge process in large deployments.

- For detailed information on partitioning, a list of the tables that can be partitioned, and how purging works in a partitioned IDB, see Purging by Truncating Partitions.
- For instructions on starting to use a partitioned IDB, see Configuring a Partitioned Oracle IDB.

Initializing IDB on PostgreSQL

To initialize IDB by running the initialization scripts provided for a PostgreSQL database:

1. Go to the directory into which you installed ICON.
2. Go to the **scripts\postgre** subdirectory.
3. Execute the following scripts in the order shown:
   1. **CoreSchema_postgre.sql**
   2. **CoreProcedures_postgre.sql**
   3. **Purge2_postgre.sql**—This script is optional. Execute this script if you want to use the **gsysPurge81** stored procedure in your deployment.
   4. **Wrapper_for_*schema version*_postgre.sql**—Execute this script if your deployment will use the **gsysIRMerge** or **gsysIRMerge2** merge procedure, or if your deployment will use the **gsysPurgeIR**, **gsysPurgeUDH**, **gsysPurgeLS**, or **gsysPurgeOS** purge procedures.
4. To execute the scripts, use the following command line to load each initialization script, providing appropriate values for the placeholders:

```
psql -h dbms_server -U user --dbname=dbname --file=script_name
```

> **Important**
>
> PostgreSQL is supported for use with Genesys Info Mart 8.x, which does not use the Interaction Concentrator merge stored procedures.

Changing IDB Instances or Re-Initializing Your Current IDB

If you reconfigure Interaction Concentrator to work with a different IDB instance, or you re-initialize your IDB, Genesys recommends that you delete the following ICON operational files before staring (or restarting) ICON:

- The Persistent Queue file. The filename is controlled by the pq-dbname option.
- The agent storage file. The filename is controlled by the agent-pstorage-name option.
- The configuration storage file. The filename is controlled by the cfg-dbname option.

Deploying DAP

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

- Generic Configuration Procedures in the *Management Framework Deployment Guide*
- *Framework DB Server User's Guide*

Configuring a Secure Connection Between ICON and Configuration Server

In the following scenario, you must configure ICON host security certificate settings at the ICON DAP `Application` object level:

- You need a secure connection between Configuration Server and ICON configured with the `cfg` role or the `all` role.
- The ICON server and the DB Server to which it is connected are on different hosts.

See Securing Connections Using TLS in the *Genesys Security Deployment Guide* for configuration instructions.

ICON-Specific DAP Deployment

In addition to the standard configuration steps, complete the following 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 the **DB Server** tab on this page).
3. Do *not* select the **JDBC Connection** check box, because it does not apply to database connections through DB Server.
4. 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 PostgreSQL, set this value to the SQL server name (usually the same as the host name of the computer where PostgreSQL runs), as configured in the DB Server Application.
     - 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 Microsoft SQL and PostgreSQL, the value is the name of the database to which the client will connect.
- **User Name**—The user name for accessing IDB. You must set a value for this property.
- **Password**—The password for accessing IDB.
- **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. This parameter 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.
  - `upper`—Field names are converted into uppercase.
  - `lower`—Field names are converted into lowercase.
  - `any`—Field names are not converted. Use the default value (`any`), unless Genesys Customer Care directs you to do otherwise.

5. When configuring a DAP Application object for IDB, do *not* configure any properties on the **JDBC Info** tab.
6. 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. To do so, perform the following steps:
   1. On the **Options** tab, create a section named **[callconcentrator]**.
   2. Within the **[callconcentrator]** section, create a configuration option named **role**.
   3. Set the option value to indicate the types of data that will be stored through this DAP. For more information, see description of role option. The **role** option values must be lower-case (for example, `cfg`). ICON interprets role option values in uppercase (`CFG`) or mixed case (`Cfg`) as invalid and defaults to the `all` role.
      - For optimal performance, Genesys recommends the following sets of values for a given database access point:
        - `gcc,gud,gls`
        - `cfg`
        - `gos`
7. 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.

Special Configuration Requirements

This section describes how to configure the Interaction Concentrator (ICON) Application object and other applications in the Genesys Configuration Layer in order to make various kinds of data available in the Interaction Database (IDB).

The following topics provide detailed instructions for storing the various sorts of data you might require:

Configuring for Voice Data
Configuring for Multimedia Data
Configuring for Attached Data
Configuring for Virtual Queue Data
Configuring for Agent State and Login Data
Configuring for Outbound Contact Data
Configuring for LRM Data
Configuring a Partitioned Oracle IDB
Configuring for High Availability
Configuring for Multi-Language Support

Configuring for Voice Data

In order to store voice interaction, agent state, and login session data in IDB, certain configuration settings are required in the Genesys Configuration Layer. This section describes the configuration settings that are required on the ICON `Application` object.

Connections:
To enable ICON to receive voice data and store it in IDB, you must configure ICON connections to appropriate T-Server instances.

This topic includes the following sections:

- Configuring for Voice Data
- Filtering Data
- Configuring DN Re-registration
- Recognizing the Correct DN in Environments Where Internal and External DNs Have the Same Name
- Hunt Group Support
- Configuring Conferencing and Transfer Options

Configuring for Voice Data

Any ICON Application object that has a configured connection to T-Server processes voice interaction data, regardless of the role that has been configured for the ICON Application. However, to enable ICON to store interaction-related and party-related data for voice calls in IDB, you must configure the role option with the value of `gcc` for the ICON application and associated Database Access Point (DAP).

To capture other types of data for voice objects and interactions, you must configure the appropriate values for the role option.

To enable ICON to identify the party that initiated release of a call, in deployments that support this functionality, set the value of the store-releasing-party option to 1.

Filtering Data

To improve Interaction Concentrator performance, consider excluding certain types of data from IDB storage. Review the filtering options in the **[filter-data]** section, and set appropriate values for your deployment.

If your deployment utilizes the feature to identify which party initiated release of a call, be aware that

certain ICON filtering options can effectively disable this functionality.

For call-based reporting, the **call-metrics** option, in the **[filter-data]** configuration section, must be set to `0` (the default). Otherwise, ICON does not write any data to the `G_CALL_STAT` table. The following options in the **[filter-data]** configuration section affect storage of information in the `G_PARTY_STAT` table:

- acd-party-history
- acd-party-metrics
- external-party
- observer-party

If you want to implement DN-based reporting on the parties that initiated release of calls, Genesys recommends that you retain the default values for these options, so that you do not filter party information.

For more information about using this feature, see How ICON Works.

Configuring DN Re-registration

Interaction Concentrator release 8.1.503.03 and higher supports multiple DN registration attempts if the initial attempt to register a DN returns an error or no response.

In earlier releases, when ICON starts up, it activates a *registration timer* and sends TregisterAddress requests to T-Server/SIP Server for each supported DN. ICON can receive in return either EventRegistered, EventError, or no response. When ICON receives EventError it marks that DN as *unregistered* and never tries to re-register that DN. Re-registration occurs only if ICON reconnects to the T-Server/SIP Server.

The new functionality enables you to automatically perform re-registration of *unregistered* DNs. To support this functionality, this release adds the gcti-re-registration-tout Switch-level and the associated gcti-re-registration-tout ICON `Application`-level configuration option. The option descriptions includes more detailed information explaining how the functionality works and the settings to use depending on your environment.

Recognizing the Correct DN in Environments Where Internal and External DNs Have the Same Name

In releases prior to 8.1.508.09, if both a DN on an internal, monitored Switch and a DN on an external Switch have the same name, ICON might incorrectly identify the otherDN as the internal DN, and therefore report incorrect data in IDB. For example, if a call went to the external DN, instead of storing the correct value of `0` in EndPointID field in related tables, ICON might store the DBID of the local DN that has the same name as the external DN. This can occur in both inbound and outbound scenarios.

To avoid this issue, you can set the same-dn option to 1, which instructs ICON to use the DN-related attributes from the EventCallPartyAdded event to collect data about the new DN. EventCallPartyAdded provides the following attributes:

- AttributePartyType indicates the party type.
  - Type = 0: internal party.
  - Type = 1: external party.
- AttributeDN indicates the device name where the new party was created.

Data Processing Details

- If you set same-dn to 1, ICON delays processing internal party data as long as the otherDN type is unknown. But although party processing may be pending, user data processing is done with no delay. As a result, there may be some differences in the cseq values.
- If ICON does not recognize the DN identified as otherDN, ICON handles it as external.
- If all DNs on both switches are unique, then setting **same-dn** to 1 does not affect party processing when otherDN is external. Results are the same as setting **same-dn** to 0. However, if all parties are internal, setting **same-dn** to 1 delays party processing in comparison to setting **same-dn** to 0.
- Even when **same-dn** is set to 1, ICON does not support scenarios in which two consecutive DN-related events are pending for the same interaction. For example, you might have an Inbound call from DN2 that is queued on DN1, DN3, and so on. The events received would be handled as follows:

	CallPartyAdded	DN1	
	Queued	DN1, DN2	If DN2 is found (that is, it is registered on the link), then event processing is pending. DN2 type is unknown yet.
	Queued	DN3, DN2	Error. The second event can not be processed as pending.

Hunt Group Support

Starting in release 8.1.504.04, Interaction Concentrator supports parallel and sequential Hunt Group reporting. Hunt Group reporting is supported only for standalone SIP Server deployments.

- For a full description of Hunt group functionality and configuration, see Hunt Groups in Standalone Deployments in the *Supplement to SIP Server Deployment Guide*.
- For information on how Interaction Concentrator records Hunt Group data in IDB, see the Interaction Concentrator 8.1 Release Note entry for release 8.1.504.04.

Configuring Conferencing and Transfer Options

A number of ICON configuration options control aspects of transfer and conference processing. This section lists relevant options and briefly explains what each does.

Options configured in the Interaction Concentrator Application object

- advanced-ext-party-reconstruction: The following are examples of call scenarios for which you might need to use this option to have Interaction Concentrator to reconstruct the external party on the unmonitored site:
  - Single-step transfer to an external number.
  - Single-step transfer to a Routing Point, which then routes the call to an external number.
- use-nts-call-state: Supports reporting on transfers made by agents using Network Attended Transfer (NAT).
- cseq-adjustment: Improves sequence tracking for user data in the G_USERDATA_HISTORY table, which enables downstream reporting applications, such as Genesys Info Mart, to correctly associate user data with interaction activity when user data updates occur within the same second that the call is transferred or terminated.
- ssc-processing: Enables support for single-step conference reporting.

Options configured on the Switch object

- gls-improve-data-for-agent: Specifies when ICON should process agent states data in two-step transfer and conference scenarios.
- lookup-queue-on-ringing: Enables ICON to identify the parent party from AttributeThisQueue in the EventRinging TEvent in complex transfer scenarios when the transfer is completed to a distribution DN before the call rings on the target DN, such as in a two-step blind transfer from a Routing Point or a Queue.
- ring-divert: Controls whether ICON identifies the PARENTPARTYID and the PARENTLINKTYPE of the Ringing party in event flows in which EventRinging comes before EventDiverted or the call is routed to an external switch.
- 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.

Configuring for Multimedia Data

In order to store multimedia interaction, agent state, and login session data in IDB, certain configuration settings are required in the Genesys Configuration Layer. This section describes the configuration settings that are required in the ICON `Application` object.

For more information about multimedia data in Interaction Concentrator, see Integrating with eServices and 3rd Party Media.

Configuring the ICON Application Object for Multimedia

To enable ICON to receive eServices and 3rd Party Media data and store it in IDB, you must configure ICON connections to appropriate Interaction Server instances.

Special Connection Procedure for Releases 8.1.500.04 and Earlier

In releases prior to 8.1.502.04, ICON cannot connect directly to an Application object of type Interaction Server. Instead, an Application object of type T-Server must represent Interaction Server.

You must perform the Interaction Server configuration differently depending on your environment, as follows:

- In a single-tenant environment or an environment with a single Interaction Server for each tenant, create a single application of type T-Server for each Interaction Server.
- In an environment with an Interaction Server that serves multiple tenants, you must create for each Interaction Server:
  - One application of the Interaction Server type (which can accommodate multiple Tenants).
  - As many applications of the T-Server type as there are tenants served by this Interaction Server, one for each tenant. Configure these applications using the actual Interaction Server host and port settings, following the instructions below.

To have ICON recognize and connect to Interaction Server, execute the following steps:

1. Create an Application with application type T-Server in Configuration Server. This might be either in addition to or instead of an Interaction Server-type Application, depending on your environment (see the explanation in the bullet-points above).
2. On the **Server Info** tab of this application, specify the host and port parameters for your Interaction Server. If you are using both an Interaction Server-type Application and a T-Server-type Application, the host and port parameters must be identical.
3. Designate the multimedia switch that Interaction Server uses as a switch for the T-Server-type Application.
4. Add the T-Server-type `Application` to the **Connections** tab of the ICON `Application` (instead of the Interaction Server `Application`).

5. On the **Options** tab, configure, at minimum, the following options:
   - Set an appropriate value for the role configuration option.
   - To enable ICON to store information about 3rd Party Media interactions in IDB, configure the mcr-om-processing configuration option.
6. If you are running ICON 8.1.0 or 8.1.1 and ICON is already started, restart it. If you are running a later version of ICON, it automatically updates the connection to the new T-Server; you do not need to restart ICON.

> **Important**
>
> During an automatic reconnection, while ICON disconnects from the prior host/port and connects to the new one, there might be a brief gap in data received from the T-Server.

Configuring for Attached Data

Attached data refers to the interaction-related business data that is sent by T–Server or Interaction Server as key–value pairs (KVPs) in the UserData, Extensions, or Reasons attributes in TEvents.

Configuring Interaction Concentrator to store attached data in IDB is a two–part process:

1. Specify the attached data key configuration file, which maps the key–value pairs (KVPs) in reporting event attributes to IDB tables and fields. For more information, see Attached Data Specification File.
2. Specify the attached data configuration settings in the Genesys Configuration Layer.

- For more information about attached data in Interaction Concentrator, see Processing Attached Data.
- For information about configuring Interaction Concentrator to store user data from EventUserEvents that are distributed by T–Server or Interaction Server from other client applications (for example, from an agent desktop application), see Configuring for Agent State and Login Session Data.

To use attached data, follow the configuration instructions in the following sections:

- ICON `Application` Object
- Attached Data Specification File
- Parser Limitations
- Attribute Values
- IDB Fields
- Universal Routing Server Attached Data

ICON Application

This section describes the configuration settings that are available on the ICON Application object.

ICON Role Configuration Option

For every ICON instance that must store attached data, make sure that the **role** option on the **Options** tab of the ICON Application object includes gud in the list of values. If you deploy a single ICON instance for the entire contact center, you can keep the default value (`all`). For more information, see the description of the role configuration option.

Attached Data Specification File

The attached data specification file (named **ccon_adata_spec.xml** by default) maps the key-value pairs (KVPs) in reporting event attributes to IDB tables and fields.

The adata-spec-name configuration option] enables you to point ICON to your attached data specification file. In ICON versions prior to 8.1.5, you must restart ICON after making changes to the

attached data specification file.

Starting in release 8.1.5, if you update the attached data specification file, ICON reads the changes dynamically. Each version of the attached data specification file is saved as a *dictionary*. By default, ICON can store up to twelve dictionaries.

Attached Data Configuration Options

The following ICON configuration options, enable you to specify what attached data ICON should store, and in what manner:

- adata-default-storage
- adata-extensions-history
- adata-reasons-history
- adata-spec-name
- adata-userdata-history
- cseq-adjustment
- max-userdata-length
- suppress-user-data
- trim-broken-utf8

Select the appropriate values for your environment, and make related configuration changes on the **Options** tab of the ICON `Application` object.

Custom Dispatcher Configuration Options

The following ICON configuration options enable you to specify how the custom dispatcher will process attached data:

- gud-cust-disp
- gud-cust-disp-groups

Review the descriptions and values for the configuration options, select the appropriate values for your environment, and make related configuration changes on the **Options** tab of the ICON `Application` object.

Attached Data Specification File

If you require ICON to store attached data in IDB, create an attached data specification for ICON to use. The attached data specification is an XML file stored in the installation directory that you specify when you install the Interaction Concentrator application.

- For the XML schema definition for your attached data specification, see the **Schema Definition** tab on the Attached Data Specification File page. This page provides the following sample attached data specifications:

- Sample Basic Attached Data Specification
- Sample Specification for Multimedia Attached Data
- Sample Specification for Customized Attached Data

In this section, as in the rest of this document, *italics* indicate placeholder text.

Parser Limitations

The ICON XML parser imposes the following limitations:

- ICON ignores unknown attributes if they are present in the specification. When parsing the XML specification, ICON checks only for missing attributes.
- The ICON XML parser does not support namespaces.

Attribute Values

This section describes the attributes that are used in the XML schema definition.

History Types

The following values can be used as history types:

`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. This value applies only to keys that are configured to be stored in the history tables.

Storage Types

The table below shows the IDB table in which each attribute is stored.

Attribute Name	IDB Table Name
`public`	G_USERDATA_HISTORY
`secure`	G_SECURE_USERDATA_HISTORY
`call`	G_CALL_USERDATA
`call-cust`	G_CALL_USERDATA_CUST

Attribute Name	IDB Table Name
call-cust1	G_CALL_USERDATA_CUST1
call-cust2	G_CALL_USERDATA_CUST2
mcr-f	GM_F_USERDATA
mcr-l	GM_L_USERDATA
user_supplied_name, such as cust-disp-group-n	Customer-defined name, as specified in the custom dispatcher.

Data Source Types

The following table shows the TEvent attribute from which each attribute is derived.

Attribute Name	TEvent Attribute Name
reasons	AttributeReasons
extensions	AttributeExtensions
userdata	AttributeUserData

IDB Fields

The mapping between the field attribute (the logical key name) in the attached data specification and fields in the IDB tables is predefined. This section describes the predefined IDB fields for:

- Voice attached data
- Multimedia-specific attached data
- Custom attached data

Predefined IDB Columns—Voice

For voice calls, the table below shows the predefined IDB field in which each attribute is stored in the G_CALL_USERDATA table.

Attribute Name	G_CALL_USERDATA Field
customer-segment	G_CUSTOMER_SEGMENT
service-type	G_SERVICE_TYPE
service-subtype	G_SERVICE_SUBTYPE
business-result	G_BUSINESS_RESULT
customer-id	CUSTOMER_ID
transaction-id	TRANSACTION_ID
cause-id	CAUSE_ID

Attribute Name	G_CALL_USERDATA Field
account-id	ACCOUNT_ID
destination-id	DESTINATION_ID
target-id	TARGET_ID

Predefined IDB Columns—Multimedia

For eServices and 3rd Party Media interactions, the following table shows the predefined IDB fields in the GM_F_USERDATA and GM_L_USERDATA tables in which multimedia-specific attributes are stored. All the IDB fields listed in this table can be used for customer-defined keys.

Important

- In this table, the **Key Name** and **Field** columns refer to Attached Data Specification attributes.
- For any field attributes marked with an asterisk (*), if it is not mapped to a customer-defined key in the attached data specification file, the IDB field will be populated with the value of the predefined key.

Predefined Key Name	Key Name	Field	IDB Field
GM_F_USERDATA Table			
FromPersonal	MyKeyName (customer-defined)	*mcr-from-name	G_FROM_NAME
	MyKeyName (customer-defined)	mcr-called-back	G_CALLED_BACK
Subject	MyKeyName (customer-defined)	*mcr-subject	G_SUBJECT
Origination_Source	MyKeyName (customer-defined)	*mcr-origin-source	G_ORIGIN_SOURCE
FromAddress	MyKeyName (customer-defined)	*mcr-from-address	G_FROM_ADDRESS
	MyKeyName (customer-defined)	mcr-reserved-1 through mcr-reserved-4	G_RESERVED1 through G_RESERVED4
GM_L_USERDATA Table			
	MyKeyName (customer-defined)	mcr-suggested-response	G_S_RESPONSE
	MyKeyName (customer-defined)	mcr-auto-response	G_A_RESPONSE
	MyKeyName (customer-defined)	mcr-auto-ack	G_A_ACK

Predefined Key Name	Key Name	Field	IDB Field
ContactId	MyKeyName (customer-defined)	*mcr-ucs-contact-id	G_UCS_CONTACT_ID

Predefined IDB Columns—Custom Fields

ICON creates the IDB `G_CALL_USERDATA_CUST*` fields in the `G_CALL_USERDATA_CUST*` tables for the custom attributes that you might use in your attached data specification. You can use these fields for both voice and multimedia interactions.

The `G_CALL_USERDATA_CUST*` fields are named `CUST_DATA_1, CUST_DATA_2, CUST_DATA_3,` and so on to `CUST_DATA_19.` Use corresponding attribute names, as shown in the following table:

Attribute Name	G_CALL_USERDATA_CUST* Field Name
CUST_DATA_1	cust-data-1
CUST_DATA_2	cust-data-2
CUST_DATA_3	cust-data-3
⋮	⋮
CUST_DATA_19	cust-data-19

Universal Routing Server Attached Data

Universal Routing Server (URS) distributes a standard set of attached data that usually exceeds reporting requirements for actual deployments.

> **Important**
> To improve performance and conserve database resources, ICON does not store values for these keys in the IDB history tables by default, regardless of the value that you specify for the adata-userdata-history option. If you require some or all of the following keys to be stored, explicitly define the respective keys in your attached data specification.

Depending on whether you specify the URS keys in the **public** or **secure** sections of the attached data specification, the KVP data is stored in the KeyName, Value, and, if you also specify the `id` attribute, KEYID fields in the G_USERDATA_HISTORY or the G_SECURE_USERDATA_HISTORY table.

For an example of an attached data specification that includes URS attached data keys, see the **Basic Sample** tab on the Attached Data Specification File page.

Tables Showing How User Data is Stored in IDB

As a result of separate ICON processing, the values of the keys in Table 1 (below) are stored in the G_ROUTE_RESULT table by default. These keys come from `userdata`. You must nevertheless explicitly include these keys in the attached data specification file if you want the key values to be stored in the user data history tables.

Table 1: source="userdata"		
RTargetRuleSelected	RTargetObjectSelected	RTargetTypeSelected
RTargetAgentSelected	RTargetPlaceSelected	RRequestedSkillCombination
RStrategyName	RTenant	RVQID
RVQDBID		

ICON masks the keys in the following three tables by default and does not store them in IDB. To store any of the `userdata` keys in Table 2, the `reasons` keys in Table 3, or the `extensions` keys in Table 4, you must explicitly specify the key in the attached data specification file.

Table 2: source="userdata"		
CBR-Interaction_cost	CBR-IT-path_DBIDs	CBR-actual_volume
CBR-contract_DBIDs	RStrategyDBID	ServiceType
ServiceObjective	RVQID	RTargetObjSelDBID
RTargetRequested	RTargetAgentGroup	RTargetRuleSelected
RTargetObjectSelected	RTargetTypeSelected	RTargetAgentSelected
RTargetPlaceSelected	RStrategyName	RRequestedSkillCombination
RTenant	RTargetUsed/RTargetName	RTargetUsed/RTargetType
RTargetAgSelDBID	CustomerSegment	RTargetPlSelDBID
RRequestedSkills	RTargetPlaceGroup	RTargetCampaignGroup
RouterData70		

Table 3: source="reasons"		
RTR	CBR-Interaction_cost	CBR-IT-path_DBIDs
CBR-actual_volume	CBR-contract_DBIDs	RStrategyDBID
ServiceType	ServiceObjective	RVQID
RTargetObjSelDBID	RTargetRuleSelected	RTargetObjectSelected
RTargetTypeSelected	RTargetAgentSelected	RTargetPlaceSelected
RStrategyName	RRequestedSkillCombination	RTenant
RTargetUsed/RTargetName	RTargetUsed/RTargetType	RTargetAgSelDBID
CustomerSegment	RTargetPlSelDBID	RRequestedSkills
RTargetPlaceGroup	RTargetCampaignGroup	RouterData70

Table 4: source="extensions"		
Reasons/RTR	Reasons/ServiceType	Reasons/ServiceObjective

Table 4: source="extensions"		
Reasons/RVQID	Reasons/RTargetObjSelDBID	Reasons/RStrategyDBID
Reasons/RTargetRuleSelected	Reasons/RTargetObjectSelected	Reasons/RTargetTypeSelected
Reasons/RTargetAgentSelected	Reasons/RTargetPlaceSelected	Reasons/RTargetAgentGroup
Reasons/RTargetRequested	Reasons/RTargetUsed	Reasons/RTargetUsed/ RTargetName
Reasons/RTargetUsed/ RTargetType	Reasons/RStrategyName	Reasons/ RRequestedSkillCombination
Reasons/RTenant	Reasons/RTargetAgSelDBID	Reasons/CustomerSegment
Reasons/RTargetPlSelDBID	Reasons/RRequestedSkills	Reasons/RTargetPlaceGroup
Reasons/RTargetCampaignGroup	Reasons/RouterData70	ReportingEventSequenceNumber
Reasons	Reasons/CBR-IT-path_DBIDs	Reasons/CBR-Interaction_cost
Reasons/CBR-actual_volume	Reasons/CBR-contract_DBIDs	RTargetUsed
RTargetUsed/RTargetName	RTargetUsed/RTargetType	

Configuring for Virtual Queue Data

This section provides information about the configuration settings in the Genesys Configuration Layer that are related to virtual queue and extended route result functionality. The default configuration settings enable the storage of virtual queue data, provided that your releases of both Interaction Concentrator and URS support virtual queue functionality.

Configuration settings on the ICON Application object, the virtual queue DN object, and the Switch object enable you to manipulate virtual queue monitoring in the following ways:

- Change the storage mode of Interaction Concentrator.
- Disable monitoring and data storage for a particular virtual queue.
- Disable monitoring and data storage at the switch level—that is, for all virtual queues that belong to a particular switch.

For more information about virtual queue data in Interaction Concentrator, see Monitoring Virtual Queues and Routing Points.

To correctly collect and store virtual queue data, following the instructions in the following sections:

- Universal Routing Server
- ICON Application Object
- Virtual Queue DN
- Switch

Universal Routing Server

Although a URS release that supports virtual queue functionality is necessary in order to enable virtual queue monitoring in Interaction Concentrator, no special configuration is required on the URS side.

Beginning in release 7.6, URS provides additional information to ICON regarding the reason for routing an interaction using the AttributeReason of routing events. URS can also attach information to interactions about the targets for which it is waiting. (For more information, see Monitoring Route Results on Routing Points.) To make this information available to Interaction Concentrator for downstream reporting purposes, set the following configuration options to `true` on the URS `Application` object:

- **report_reasons**
- **report_targets**

For more information about these URS configuration options, see the *Universal Routing Reference Manual*.

ICON Application Object

The default settings enable ICON to receive virtual queue data and store it in IDB.

Connections

Although a URS release that supports virtual queue functionality is necessary in order to enable virtual queue monitoring in Interaction Concentrator, ICON receives the data from T–Server or Interaction Server. Therefore, no connection to URS is required.

vq-write-mode

The vq-write-mode configuration option enables you to switch the storage mode for virtual queue data, if necessary. For descriptions of the storage modes, see Monitoring Virtual Queues and Routing Points.

extended-route-result

The extended-route-result configuration option specifies whether ICON stores extended routing results (from URS) in IDB.

> ### Important
>
> - To store extended route results in IDB, ICON requires URS release 7.6 and Interaction Server release 7.6.000.18 (or higher).
> - Interaction Concentrator functionality related to storing Virtual Queues history in the G_ROUTE_RES_VQ_HIST table requires URS release 8.1 or higher.

For more information about these options, see the following configuration option descriptions:

- extended-route-result
- route-res-vqid-hist-enabled
- store-route-result-reliability
- vq-write-mode

Virtual Queue DN

Unless you need to disable monitoring and data storage for a particular virtual queue, no configuration is necessary for the DN object that represents this virtual queue in the Configuration Layer.

monitor

The monitor configuration option enables you to turn off ICON monitoring and data storage for a particular virtual queue, if necessary. If the option is set to `0`, ICON does not register with T-Server to receive events that pertain to this virtual queue.

Switch

Unless you need to disable virtual queue monitoring and data storage for a particular switch, no configuration is necessary for the corresponding Switch object in the Configuration Layer.

support-dn-type-N

The support-dn-type-N configuration option enables you to turn off ICON monitoring and data storage for all DNs of a specified type that belong to a particular switch. To set the type of DN to which the option should apply, create an instance of this option in which you replace the variable *N* with the number corresponding to the DN type you want to control.

For example, to turn off monitoring for all virtual queue DNs on a switch, set the type to 5, indicating that it applies to the virtual queue DN type (that is, the option name is set to **support-dn-type-5**), and set the option value to `0`. With this setting, ICON does not register with T-Server to receive events that pertain to virtual queue DNs that belong to this switch. In this case, ICON does not process or store virtual queue related TEvents, even if the monitor option is set to 1 for any of the virtual queues that belong to the switch.

Configuring for Agent State and Login Data

In order to store agent state and login session data for voice and multimedia interactions in IDB, certain configuration settings are required in the Genesys Configuration Layer.

To enable ICON to receive agent data and store it in IDB, you must configure ICON connections to appropriate T-Server and Interaction Server instances.

> **Important**
> When ICON terminates a login session as 'stuck'—that is, some issue has made it necessary to terminate the login session without receiving EventLogout—all active reason codes related to the terminated session are removed from the G_AGENT_STATE_RC_A table (stores active reason codes) and are not transferred to the G_AGENT_STATE_RC table (stores reason code history).

To store agent state and login data, use the instructions in the following sections:

- ICON Application Configuration
- Configure ICON to Use Custom States
- Start Recording a Custom State
- Send Custom State Data
- Stop Recording a Custom State
- Use Multiple Custom States at Once

ICON Application Configuration

ICON Role
For every ICON instance that must store agent state or agent login session data, make sure that the role option includes `gls` in the list of values. If you deploy a single ICON instance for the entire contact center, you can keep the default value (`all`).

Other Options
Interaction Concentrator provides a number of options to control reporting on agent login session metrics and agent login sessions:

- gls-acw-first
- gls-stats-update
- gls-stats-update-delta

- gls-store-event-seq
- gls-active-reason-codes
- gls-enforce-reason-code

Configure ICON to Use Custom States

In order for ICON to store information to support reporting about custom states and common data, you must do the following:

1. Set appropriate values for the following ICON Application configuration options:
   - AgentRecordUserTypes, which defines the custom agent states.
   - AgentUserFields
   - EventData
   - EventExtensions
   - GlobalData
   - max-party-info
   - store-event-data
   - store-event-extensions
2. Configure the agent desktop application to send the applicable key-value pairs (KVPs) to T-Server, so that they can be included in the UserData of EventUserEvent, as explained below.

Agent Desktop Application Configuration

ICON records the beginning and end of a custom state, based on information that it receives in the UserData of an EventUserEvent from T-Server. You must configure your agent desktop application to send T-Server the appropriate KVP information for the EventUserEvent UserData.

Start Recording a Custom State

In order for ICON to start recording a custom state, the desktop application must send the following KVP:

`Key = "StateKeyName", Value = "+"`

Example

```
"AfterCallWork", "+"
```

Send Custom State Data

In order for ICON to store additional information about an active custom state, the desktop application must send the following KVP:

Key = "*CommentKey*", Value = "*StateCode*, *Comment*"

You can configure more than one comment key for the same custom state. However, for each comment key, ICON can store only one value. If multiple KVPs are sent for the same comment key, ICON stores only the last value.

Example

```
"Comment", "207, This is data about the state" "Explanation", "207, This is more data about the state" "Explanation", "207, This is more, changed data about the state"
```

In this example, ICON will store the following values:

- In the **Comment** field for state 207: `This is data about the state`
- In the **Explanation** field for state 207: `This is more, changed data about the state`

Stop Recording a Custom State

In order for ICON to stop recording a custom state, the desktop application must send the following KVP:

Key = "*StateKeyName*", Value = "–"

Example

"AfterCallWork", "–"

Use Multiple Custom States at Once

For each type of custom state, only one state can be active for a DN at any one time. However, ICON can simultaneously handle multiple different states independently. For example, two different states can be active on one DN, with different data corresponding to each. ICON does not support duplicate key names in attached data; KVPs with the same key name should not be sent in one EventUserEvent (as shown in the following Example).

Example

```
"AfterCallWork", "+"
"Break", "+"
"Comment", "207, This is data about the call"
```

```
"Comment", "208, This is data about the break"
"Break", "–”
"AfterCallWork", "–"
```

In the example above, ICON will store the key value only for the custom state `"207, This is data about the call"`.

Configuring for Outbound Contact Data

In order to store Outbound Contact data in IDB for reporting purposes, certain configuration settings are required in the Genesys Configuration Layer, both for certain Outbound-related configuration objects and for the ICON `Application` object.

- Outbound Contact Configuration
- ICON Application Object
- Multi-Tenant Considerations

For more information about Outbound Contact Server (OCS) data in Interaction Concentrator, see Integrating with Outbound Contact.

Outbound Contact Configuration

Special configuration of the items listed below is required in order to enable OCS to process and send data to ICON about the content of the fields in calling list records.

- Field Object
- Campaign Group Object
- Outbound Contact Server

Field Object

The `Field`-level configuration options (described in detail below) controls whether ICON will receive and store field values:

- **icon_attribute**
- **send_attribute**

> **Important**
> Interaction Concentrator reads `Field` object configuration information only at startup. No real-time configuration changes to `Field` objects are recognized. To accept changes to `Field` configuration, restart Interaction Concentrator.

icon_attribute

For every `Field` configuration object that describes a single field (for example, a phone number)

within a record, you must configure the **icon_attribute** option if you want that data to be stored in IDB.

To configure this option:

1. Open the **Properties** dialog box for the particular Field configuration object.
2. Click the **Annex** tab.
3. Create a new section named **[default]**, if it does not already exist.
4. Within this section, create a new option named **icon_attribute**.
5. Set the option to one of the following values:
   - 1: To store OCS mandatory fields in the GO_RECORD table, custom defined fields in the GO_CUSTOM_FIELDS table, and history of field changes in GO_FIELDHIST table.
   - 2: To store data as a secured field in the special GO_SECURE_FIELDS and GO_SEC_FIELDHIST IDB tables.

If you do not configure this option, or if you set its value to `0` (zero), OCS will not deliver those fields to ICON when sending reporting information, and ICON will not store the value of such fields.

send_attribute

For every user-defined field that describes a single field (for example, a customer name) within a record, configure the Outbound Contact **send_attribute** option to have OCS attach that data to outbound calls and in user events.

> **Important**
>
> You do not need to specify mandatory fields, such as GSW_CALL_ATTEMPT_GUID and GSW_CALL_TYPE, in the send_attribute option. These fields are attached by default.

By default, OCS attaches the values of the mandatory fields listed in the table below. The table also shows the default key name for the attached data key–value pair.

Field	Key Name
contact_info	GSW_PHONE
chain_id	GSW_CHAIN_ID
attempt	GSW_ATTEMPTS
call_result	GSW_CALL_RESULT

If you do not configure the **send_attribute** option for user-defined fields, OCS does not process data that is related to those Field objects, and accordingly ICON does not receive that data.

For more information, see the description of the **send_attribute** option in the *Outbound Contact Deployment Guide*. See also the section about attaching record information to desktop and OCS user events in Communication Protocols in the *Outbound Contact Reference Manual*.

Campaign Group Object

To enable reporting for all the activity associated with a Campaign Group, including chain activities, ensure that the Campaign Group object's configuration properties specify a valid Voice Transfer Destination DN. The DN must be located on the switch served by the T-Server to which OCS is connected, and the T-Server must have a CTI link connected with the switch.

Outbound Contact Server

If you require OCS to report snapshot metrics that are based on calculations related to call times (Outbound Call Dialing Time, Outbound Call Transfer Time, and CPD Time), ensure that audit logging is enabled for the OCS `Application` object.

- To enable audit logging, set the **log_call_stats** configuration option to `true` or `yes`.

No other special configuration is required on the OCS `Application` object.

ICON Application

To enable ICON to receive OCS data and store it in IDB, you must configure ICON connections to appropriate data sources, and you must set relevant configuration options.

> **Important**
> For detailed information about deploying Outbound Contact with Interaction Concentrator, see Integrating with Outbound Contact in the *Interaction Concentrator User's Guide*.

- Connections
- ICON Role Configuration Option
- OCS–Specific ICON Options

Connections

To have ICON store Outbound Contact data, you must add at least one OCS instance to your ICON `Application` object **Connections** tab.

- In an environment with a single OCS instance, add the OCS `Application` object to the **Connections** tab of the ICON `Application` object.
- In an environment with multiple OCS instances, decide on your deployment topology—that is, decide whether a single ICON instance will handle the data from all or a subset of OCS instances, or whether each OCS will have a dedicated ICON instance. Based on your deployment decision, add one or more

OCS `Application` objects to the **Connections** tab of the ICON `Application` object that must store data from those OCS instances.

Specifying Which Switches to Use for Outbound Data

Each OCS instance can connect to multiple switches. If you simply add OCS to your **Connections** tab, ICON takes Outbound data from all switches to which the specified OCS instance(s) are connected.

To take Outbound data only from some of the total pool of switches, add T-Server(s) associated with the desired switch(es) to the ICON `Application` object **Connections** tab. ICON then takes data only from the switches that are associated with the specified T-Servers.

For a detailed discussion of these deployment considerations, see Outbound Contact Deployment Scenarios in the *Interaction Concentrator User's Guide*.

ICON Role Configuration Option

For every ICON instance that must store outbound data, make sure that the role option on the **Options** tab of the ICON `Application` object includes `gos` in the list of values. If you deploy a single ICON instance for the entire contact center, you can keep the default value (`all`).

If you store different types of data to different IDBs, make sure that `gos` is also specified for the role option on the **Options** tab of the appropriate Database Access Point (DAP). Configure this option on the **Options** tab of the `Application` object for the DAP that your ICON instance uses to store outbound data to IDB.

OCS–Specific ICON Options

The following ICON configuration options enable you to specify what outbound data ICON should store, and in what manner:

- gos-write-duplicate-metrics
- gos-write-metrics
- gos-write-metrics-only

Select the appropriate values for your environment, and then make the appropriate configuration changes on the **Options** tab of the ICON `Application` object.

Multi-Tenant Considerations

In multi-tenant environments, the OCS-related objects that the ICON instance monitors may be configured under various tenants. Ensure that you assign all related tenants to the ICON `Application`.

Multi-Tenant Example

For example, you might create an Outbound Calling List object under a tenant called **Outbound**, and have the calling list use fields that you created as Field objects under the **Environment** tenant. To enable ICON to process OCS data related to the Outbound Calling List:

1. Configure the required Field objects under the **Environment** tenant.
   - Configure the **icon_attribute** option for all the fields that you want ICON to store.
   - Configure the **send_attribute** option for all the fields that you want ICON to store.
2. Add both the **Environment** tenant and the **Outbound** tenant on the **Tenants** tab of the ICON `Application` object.

Configuring for LRM Data

If you are running Genesys License Reporting Manager (LRM), ICON enables you to store your LRM-specific data. This section explains the ICON-specific aspects of configuring ICON to work with LRM. For detailed information about LRM, see the Genesys License Reporting Manager documentation.

Configuring ICON to work with LRM requires that you set the appropriate value for the ICON role configuration option.

- The recommended approach is to use dedicated ICON instances set to the `lrm` role. If the ICON instance must be shared with reporting applications, then ICON should be set to the `gls` and `gos` roles, which can also support LRM. The `lrm` role should *not* be combined with the `cfg` role.
- If the **role** value is set to `all`, ICON stores LRM data. However, if you require *only* LRM data, setting the value to `all` results in the accumulation of large quantities of unusable data. Genesys recommends that you explicitly set the value to `lrm` to collect License Reporting data.

Configuring ICON to store LRM data

To configure ICON, perform the following steps:

1. Designate the instance of ICON that will be used for LRM.
2. Set the value of the **role** option to `lrm`.
3. Start (or if applicable, restart) ICON to have the **role** option setting take effect.

Configuration Notes

- Role assignments must be configured using only lower case (for example, `lrm`). ICON interprets uppercase (LRM) or mixed case (Lrm) settings as invalid and defaults to the `all` role.
- When **role**=`lrm`, ICON:
  - Does not write data for `gcc` or `gud` providers.
  - Disregards data filtering options.
  - Enforces having the use-dss-monitor option set to `true`.
- You can switch ICON to or from the `lrm` role at any time by changing the setting for the **role** option. Restart ICON to have the change take effect.

Important

It is not advisable to change roles without careful planning. ICON stores the data associated with a role only when it is configured with that role. For example, if you set an instance of ICON to collect LRM data, then change the role so it is no longer set to `lrm`, and then later change it back again, you will probably have a window of time during which there is no LRM data stored because the previous role may not have

required ICON to collect the data necessary for LRM reporting.

- If you are using Genesys Info Mart, do not try to connect it to the ICON and the associated IDB that stores the LRM data. The LRM–specific ICON–IDB set stores data in a specific subset of tables. As a result, Genesys Info Mart will fail to start when it finds the tables from which it extracts data to be empty.
- HA is supported just as for any other ICON role.

Configuring a Partitioned Oracle IDB

In Oracle environments with large amounts of data to maintain, you can choose to create a partitioned Oracle IDB, which you can then purge efficiently by truncating entire partitions using the **purgePartitions811** stored procedure. During this purge, all records in the purged partitions—both terminated and non-terminated—are truncated unconditionally.

> **Important**
>
> - If you need to purge only non-terminated records, use the **GSYSPurge80/ gsysPurge81** purge procedure.
> - This partitioning and purge functionality is supported only for Oracle 11g and higher.
> - Genesys strongly recommends that you do not use this purge mechanism for long-lived data types, such as multimedia. When used with long-lived data types, you might encounter situations in which some of the data for a still-active interaction is purged.

- Overview
- Creating Your Database
- About Partitioning
- Purging
- Partitioned Tables

Overview

The procedure for deploying the purge-by-truncating-partitions functionality is outlined below.

1. Start with a new Oracle database. There is no migration from a non-partitioned to a partitioned IDB.
2. Run the SQL scripts to create your partitioned IDB, using the standard procedure given in Deploying IDB. However, the scripts used to create a partitioned IDB differ from those used for nonpartitioned IDBs, so be sure to see “Creating Your Database” (below) for a list of the correct scripts.

Creating Your Database

To create a partitioned Oracle IDB, follow the standard instructions for Deploying IDB, but run the following scripts rather than the scripts used for a nonpartitioned IDB. These two initialization scripts

create a new partitioned IDB:

- **CoreSchemaPart_ora.sql**
- **CoreProcedures_ora.sql**

The following initialization script sets up the stored procedure used to purge the partitioned Oracle IDB:

- **PurgePart_ora.sql**

As noted above, there is no migration path from a nonpartitioned to a partitioned database.

About Partitioning

The number of partitions is fourteen, with each partition equivalent to one day. Data is written into the partitions in sequence, starting with Partition 1 on Day 1, Partition 2 on Day 2 and so on, circling back to Partition 1 on Day 15.

As with all purge methods, only operational tables are purged. Special tables used for internal data storage and retrieval are neither partitioned nor purged.

The tables that are available for partitioning include the **gsys_partition** field, which must be configured to contain the UTC value taken from the **created_ts** field. This parameter is set using the partition-type configuration option.

Each partitioned table also includes the virtual GSYS_SHORT_DAY column, based on value of the **gsys_partition** field.

Purging

You perform the purge by executing the **purgePartitions811** stored procedure, which truncates all partitions except for [the number you specify in the `days-to-keep` parameter of the SQL statement] + [an additional safe-guard or *tomorrow* day].

Instructions for how to run the **purgePartitions811** procedure, how to schedule it, and all other operational considerations are documented in Purging by Truncating Partitions.

Partitioned Tables

The following tables are partitioned by the **CoreSchemaPart_ora.sql** script:

G_AGENT_STATE_HISTORY	G_IR_HISTORY	GO_CHAINREC_HIST

G_AGENT_STATE_RC	G_IS_LINK	GO_CUSTOM_FIELDS
G_CALL	G_IS_LINK_HISTORY	GO_FIELDHIST
G_CALL_HISTORY	G_LOGIN_SESSION	GO_METRICS
G_CALL_STAT	G_PARTY	GO_RECORD
G_CALL_USERDATA	G_PARTY_HISTORY	GO_SEC_FIELDHIST
G_CALL_USERDATA_CUST	G_PARTY_STAT	GO_SECURE_FIELDS
G_CALL_USERDATA_CUST1	G_ROUTE_RESULT	GOX_CHAIN_CALL
G_CALL_USERDATA_CUST2	G_USERDATA_HISTORY	GS_AGENT_STAT
G_CUSTOM_DATA_P	G_VIRTUAL_QUEUE	GS_AGENT_STAT_WM
G_CUSTOM_DATA_S	GM_F_USERDATA	GX_SESSION_ENDPOINT
G_CUSTOM_STATES	GM_L_USERDATA	G_ROUTE_RES_VQ_HIST
G_DND_HISTORY	GO_CAMPAIGNHISTORY	G_SECURE_USERDATA_HISTORY
G_IR	GO_CHAIN	GO_CAMPPROP_HIST

Configuring for High Availability

The High Availability (HA) model used in Interaction Concentrator differs significantly from the Genesys standard HA model implemented in a majority of Genesys servers. Before you configure your ICON HA deployment, review The Interaction Concentrator HA Model in the *Interaction Concentrator User's Guide*.

In an HA deployment, observe the following rules:

- You must set configuration options in both Interaction Concentrator `Application` objects exactly the same. Because this is not a typical redundant pair from the Genesys perspective, Configuration Server does not automatically synchronize the configuration options for two ICON applications.
  For example, to configure your redundant ICON `Applications` to store voice interaction data in a pair of HA IDBs:
  - In both ICON `Application` objects, set the role configuration option so that it contains `gcc` and `gud`. This enables both ICON `Applications` to store call related and attached data.
  - For any configuration options that affect the data populated by those roles, set the same option values in both ICON `Applications`. For example, the two applications must use the same ICON configuration options for virtual queue monitoring, storage of attached data, and so on.
- For more information about setting configuration options, refer to the other pages in this section.
- You must configure a connection to the same T-Server or Interaction Server in both ICON `Application` objects.
- You must create two identical IDBs. Genesys recommends using two databases located on different hosts, but having the same RDBMS type and version number, to host the HA pair of IDBs.
- You must configure a DAP for each ICON to access its IDB.

For more information about configuring applications and connections, see the *Management Framework Deployment Guide*.

Configuring for Multi-Language Support

For multi-language support, the following items might apply depending on your environment and requirements:

- If you require UTF-8 support, you must use Configuration Layer components of release 8.1.3 or higher.
- If you are running an Oracle database, make sure the NLS_LANG environment variable on the host where DB Server is installed is set to match the character encoding of data in the Oracle database; for example, `NLS_LANG=.UTF8`. This is especially important when using UTF-8 language encoding, and in all Chinese, Japanese, and Korean language environments. For additional information on specific configuration information for Oracle databases, see the Framework Database Connectivity Reference Guide.
- In environments with PostgreSQL RBDMS, the IDB processing configuration details (having the role option set to `cfg`) should be created to use encoding identical to that used by Configuration Server. For example, you may set PostgreSQL database encoding to `WIN1252` to use extended ASCII from European languages, or use SQL_ASCII.
- In environments with PostgreSQL RBDMS, if you require ICON to process user data using any symbols other than ASCII, you must create your PostgreSQL IDB with encoding that supports these symbols. If this encoding is different from the Configuration Server encoding, you must have a separate IDB for configuration details.
- When storing long Unicode string values in a multi-language environment, keep in mind the following points:
  - **WARNING! Genesys does not support any user edits to the IDB schema.** Changing the schema or trying to increase the length of a field can cause unpredictable data loss.
  - To avoid database errors when inserting string values longer than the matching table field, ICON truncates string data that exceeds the specified field length. See the description of fields in the Interaction Concentrator Physical Data Model document for your RDBMS type for the length set for each.
  - For user attached data, ICON truncates strings longer than the value set in the max-userdata-length option.
  - To avoid incomplete UTF-8 symbols at the end of truncated string, configure the trim-broken-utf8 option.

Unicode Support

How you configure Interaction Concentrator for Unicode support depends on which RDBMS you are using for IDB.

- Microsoft SQL—See the detailed configuration instructions in Configuring for Unicode Support in an Environment with a Microsoft SQL IDB below.
- Oracle and PostgreSQL—No special configuration of Interaction Concentrator is required. Note, however, that your database itself must be configured to support Unicode.
- DB2—Unicode is not currently supported.

Configuring for Unicode Support in an Environment with a Microsoft SQL IDB

Starting from release 8.1.510.07, Interaction Concentrator supports storage of Unicode user data on Microsoft SQL IDBs. The affected fields use the nvarchar datatype instead of the varchar datatype. Note that as a result, these fields use twice as much space as they would if they used the varchar datatype.

Important

- If you require Unicode support, you must use DB Server 8.1.301.11 or higher.
- International symbols will be written correctly only if ICON receives them in Unicode (UTF-8). To use Unicode, it is mandatory to have UTF-8 encoding configured in Configuration Server, in your XML attached data specification, and in all clients that attach international data to interactions.

The Unicode Microsoft SQL database schema uses nvarchar fields for all columns that store data entered by the user. To configure Interaction Concentrator on a Microsoft SQL IDB for Unicode, perform the following steps:

1. Install DB Server 8.1.301.11 or higher. Configure DB Server with the following settings in the **[dbserver]** configuration options section:
   - **msql_name** = `./dbclient_851/dbclient_msql.exe`

   For additional information about DB Server, see the *DB Server User's Guide*.
2. Set up your Configuration Server for Unicode support (that is UTF-8 encoding). For instructions, see Deploying Configuration Server in the *Framework Deployment Guide*.
3. Verify that your other data source applications are configured for Unicode support. For instructions, see the relevant deployment information for T-Server, Interaction Server, Outbound Contact Server.
4. If you are using Unicode in your attached data, ensure that your XML attached data specification is saved with UTF-8 encoding. (If all of your defined data keys are in ASCII format, you can save the XML attached data specification file in ASCII format.)
   - For more on setting up your attached data specification, see Configuring for Attached Data.
5. Create a new IDB. There is no migration from an existing IDB to Unicode. To do so:
   - Run the **CoreSchema_multilang_mssql.sql** database initialization script (rather than the **CoreSchema_mssql.sql** initialization script) on the new database.
6. Set the appropriate value for the support-unicode ICON `Application` option.
7. Check the ICON log file for the following messages indicating whether there are configuration issues in your environment:
   - 09-25032
   - 09-25033
   - 09-25034

- 09-25035
- 09-25036

If you have questions about this process, please contact Genesys Customer Care.

Additional Notes

There is no migration path for an existing Microsoft SQL IDB to a Unicode one. If you need special help with data transfer, contact Genesys Customer Care for assistance.

The Interaction Concentrator upgrade procedure remains unchanged. In case of future database schema changes, a single upgrade script will upgrade both Unicode and non-Unicode IDBs.

Configuration Options

> **Important**
>
> The options described on this page have been moved into the *Interaction Concentrator Options Reference*. The links below take you to the specified sections within the *Options Reference*.

Interaction Concentrator-related options are set in the following `Application` objects:

- Interaction Concentrator
- Switch
- DN
- Script
- DAP

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

Interaction Concentrator Options

> **Important**
>
> The options described on this page have been moved into the *Interaction Concentrator Options Reference*. The links below take you to the specified sections within the *Options Reference*.

- **[callconcentrator]** Section
- **[custom-states]** Section
- **[dbw-error-reactions]** Section
- **[filter-data]** Section
- **[listeners]** Section
- **[user_named_section]** Section
- **[log]** Section

ICON Configuration Options, [callconcentrator] Section

> **Important**
>
> The options previously described on this page have been moved into the **[callconcentrator]** Section of the *Interaction Concentrator Options Reference*.

Switch Options

> **Important**
>
> The options previously described on this page have been moved into the **[Switch]** Section of the *Interaction Concentrator Options Reference*.

DN Options

> **Important**
>
> The options previously described on this page have been moved into the **[DN]** Section of the *Interaction Concentrator Options Reference*.

Script Options

> **Important**
>
> The options previously described on this page have been moved into the **[Script]** Section of the *Interaction Concentrator Options Reference*.

DAP Option

> **Important**
>
> The options previously described on this page have been moved into the **[DAP]** Section of the *Interaction Concentrator Options Reference*.

Starting and Stopping

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.

This topic includes the following information:

- Before Starting
- Command-Line Parameters
- Starting Interaction Concentrator
- Stopping Interaction Concentrator

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

Before Starting

The following issues are important for you to consider before you attempt to start ICON.

Starting the cfg Role for an Oracle IDB

If the relational database management system (RDBMS) of the Interaction Database (IDB) that stores configuration-related data is Oracle, Genesys strongly recommends that you collect statistics on your IDB schema before you start the ICON instance that performs the `cfg` role. Collecting statistics before starting ICON significantly shortens the amount of time it takes to start up.

> **Tip**
> This applies only to the second and subsequent start-ups. The first time you start ICON in a new installation, the IDB is empty.

Verifying ICON Connections and Configuration

Before you attempt to start ICON, confirm that the connections and configuration options that have been configured for your ICON `Application` are correct for your deployment.

Connections

In general, do not change any connections on the **Connections** tab of the ICON `Application` during startup or runtime. Furthermore, do not disconnect from Configuration Server during startup.

- If ICON disconnects from Configuration Server during startup, ICON initialization will fail.
- If you remove or change other connections during startup, ICON might fail to initialize correctly.
- If you remove or change any connections during runtime, ICON functioning might be affected.

> **Important**
>
> You must restart Interaction Concentrator after a backup instance is configured of any application, such as OCS or T-Server, for which Interaction Concentration has a connection configured on the **Connections** tab. If you do not restart Interaction Concentrator, data from the affected application is not written to the database.

For more information about configuring connections, see the instructions for configuring the **Connections** tab.

Configuration Options

Do not make changes to ICON configuration options during startup. You can make changes to ICON configuration options during runtime, but in some cases you must restart ICON for the changes to take effect. For more information, see Interaction Concentrator Options Reference.

Command-Line Parameters

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

Parameter	Description
`-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.
`-transport-port`	The client-side port that ICON must use to connect to Configuration Server.
`-transport-address`	The TCP host address to be used for the transport-port.
`-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

Parameter	Description
	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 Interaction Concentrator

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

- Starting ICON with Solution Control Interface
- Starting ICON manually on UNIX
- Starting ICON on Windows
- Starting ICON as a Windows Service

Starting ICON with Solution Control Interface

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

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 sources.

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.)

> **Tip**
>
> 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:

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

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

Starting ICON manually on UNIX

Complete the following procedure to start ICON manually on UNIX.

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 host_name -port port_number -app application_name
```

Where:

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

> **Important**
> 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"
```

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.

To start ICON from the **Start > Programs** menu, or from the console window.

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

> **Important**
> 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"
```

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.

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 is not available.)

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.

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

Stopping Interaction Concentrator

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

- Stopping ICON with Solution Control Interface (Recommended)
- Stopping ICON on UNIX from the command line
- Stopping ICON on UNIX from the console window
- Stopping ICON on Windows from the console window
- Stopping ICON as a Windows Service

Stopping ICON with Solution Control Interface (Recommended)

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

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

Stopping ICON on UNIX from the command line

Stop ICON on UNIX by using the following procedure.

- On the command line, enter the following:
  `kill -SIGTERM processid`

  Where *processid* is the application's UNIX process ID.

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

Stopping ICON on UNIX from the console window

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

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

Stopping ICON on Windows from the console window

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

- From the application's console window, press **CTRL+C.**

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

Stopping ICON as a Windows Service

If you are running ICON as a Windows Service, you should stop it only from the Services Control Manager. To stop Interaction Concentrator running as a Windows Service, use the following 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.**

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

Troubleshooting

This section describes problems that you might encounter when starting or running your Interaction Concentrator (ICON) application, and how to resolve them.

- Startup Problems
- Runtime Problems
- Merge Problems
- Purge Problems

> **Warning**
> To avoid a wide range of startup and runtime problems, observe the following restrictions:
>
> - Do not disconnect ICON from Configuration Server during startup.
> - Do not change any connections on the **Connections** tab of the ICON `Application` during runtime.

Startup Problems

The following are the most common startup problems:

- ICON does not connect to the Configuration Server
- ICON Exits at Startup

ICON does not connect 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.

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.

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.

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 Deploying DAP.

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.

> **Important**
> If you make changes to the attached data specification file, you must restart ICON for them to take effect.

The persistent queue (pq) 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.
    With either method, 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 Populating Agent Login Session Data.

Runtime Problems

The following are the most common runtime problems:

- ICON does not connect to T-Server or Interaction Server
- ICON Does Not Receive Call-Related Events from T-Server
- ICON Does Not Write Information to the Database
- CON Has Lost Synchronization with the Configuration Database
- ICON Does Not Function Correctly

ICON does not connect 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. Interaction Server release 7.5 is the minimum version required for Interaction Concentrator support of eServices. For more information about Interaction Concentrator compatibility and interoperability with other Genesys components, see "Compatibility" on the Planning Your Deployment page.

- **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.

ICON was not restarted after a backup instance was configured of a T-Server to which Interaction Concentrator has a connection configured on the **Connections** tab.

- **Solution:** Stop ICON, then restart.

There is no connection between the ICON `Application` object and T-Server.

- **Solution:** 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 Deploying DAP.

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 Deploying DAP.

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

- **Solution:** Run the Interaction Concentrator initialization scripts. For more information, see Deploying IDB.

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

- **Solution:** Stop ICON, then restart.

ICON was not restarted after a backup instance was configured of a T-Server to which Interaction Concentrator has a connection configured on the **Connections** tab.

- **Solution:** Stop ICON, then restart it.

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 see the following options: acc-proc-tout, acc-queue-

lifespan, and acc-queue-size.

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 an ICON release earlier than 8.1.512.08 and have configured an HTTP Listener, you can also view the error messages on the Database Writer performance counter web page (for more information, see Monitoring Interaction Concentrator in the *Interaction Concentrator User's Guide*).

> **Important**
>
> From release 8.1.512.08 forward, the HTTP Listener functionality is no longer supported.

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.

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. For more information, see Resynchronizing Configuration Changes in the *Interaction Concentrator User's Guide*.

ICON Does Not Function Correctly

Possible causes of this problem are as follows:

A connection configured on the **Connections** tab of the ICON `Application` object was removed or changed while ICON was operating.

- **Solution:** Stop ICON. Verify that the connections that have been configured on the **Connections** tab of the ICON `Application` object are as required for the deployment, then restart ICON. For more information about configuring connections, see Configure the Connections tab.

Merge Problems

> **Important**
>
> - The merge stored procedure described on this page is necessary only if you are running Genesys Info Mart 7.6 or earlier or you are running Interaction Concentrator *without* Genesys Info Mart. Genesys Info Mart provides a merge procedure that supplants the one documented on this page.
> - This merge procedure is not supported on PostgreSQL RDBMSs.

For more information about the merge procedure, see Merge Stored Procedure in the *Interaction Concentrator User's Guide*.

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

- Merge Procedure Does Not Complete Successfully
- Merge Procedure Does Not Execute
- Merge Procedure Performance Is Slow or Unstable

Merge Procedure Does Not Complete Successfully

Possible causes of this problem are as follows:

In MS SQL Server 2019 Cluster deployments, switchover to another host in the cluster occurred while the merge procedure was executing. The error message is similar to the following:

```
Remote harden of transaction 'implicit_transaction'...failed.
Msg 596, Level 21, State 1, Line 0
Cannot continue the execution because the session is in the kill state.
Msg 0, Level 20, State 0, Line 0
A severe error occurred on the current command. The results, if any, should be discarded.
...
```

- **Solution:** Restart the procedure.

In general, the most likely reason the merge procedure fails is an inconsistency in IDB. The database

inconsistency might be introduced by ICON, by the downstream reporting application, through manual intervention, or in some other way. For example, if ICON writes a duplicate G_IS_LINK record while the merge procedure is executing, the RDBMS might report a primary key violation. Describing the possible causes of this problem in detail is beyond the scope of this document.

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

- GSYS_PENDING_IR
- GSYS_PENDING_LINK
- GSYS_SYSPROCINFO

- **Solution:** Review the error messages reported in the ICON log file, and take appropriate action to resolve the cause of the failure. You might also have to reset the merge procedure so that it recovers from its failed state (see Merge Procedure Recovery). Then restart the merge procedure.

Merge Procedure Recovery
Interaction Concentrator provides a stored procedure, **gsysIRMergeReset**, to simplify the steps to reset the merge procedure to recover from a failed state. To invoke the procedure, use an SQL statement like the following (the exact syntax depends on the RDBMS):

```
EXEC gsysIRMergeReset
```

> **Important**
> 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, execute the stored procedure to reset the merge 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 the execution command as required. For more information, see Executing the Merge Procedure in the *Interaction Concentrator User's Guide*.

There is an error in the database or in database performance that is not specifically related to the merge procedure or to ICON—for example, insufficient disk space or insufficient privileges.

- **Solution:** Review the error messages reported in the ICON log file. Provide the appropriate fix that the RDBMS requires, then restart the merge procedure.

The database error might be related to an inconsistency in IDB, in the sense that it was exposed or induced by an inconsistency in IDB, or resulted in an inconsistency in IDB. In these cases, reset the merge procedure (see Merge Procedure Recovery), then restart the merge procedure. If the merge procedure still fails to execute, contact Genesys Customer Care.

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:** Contact Genesys Customer Care for assistance with database locking issues.

There is an inconsistency in IDB that does not cause the merge procedure to fail, but that significantly interferes with merge procedure performance.

- **Solution:** Reset the merge procedure (see Merge Procedure Recovery), then restart the merge procedure. If the problem persists, review database settings and try general database tuning adjustments. If the problem still persists, contact Genesys Customer Care.

Purge Problems

For more information about the purge procedures, see Purge Procedures in the *Interaction Concentrator User's Guide*.

The most common problems encountered in executing the purge procedures are as follows:

- PurgeProcedure Does Not Complete Successfully

Purge Procedure Does Not Complete Successfully

Possible causes of this problem are as follows:

In MS SQL Server 2019 Cluster deployments, switchover to another host in the cluster occurred while the purge procedure was executing. The error message is similar to the message shown above for the merge procedure.

- **Solution:** Restart the procedure.

Attached Data Specification File

This section presents the XML schema definition for processing key-value pairs (KVPs) from the attached data that T-Server or Interaction Server, if applicable, provides with TEvents. If you require ICON to store attached data in IDB, create an attached data specification for ICON to use, based on the information in this section.

- About the Attached Data Specification File
- Schema Definition
- Basic Sample
- Multimedia Sample
- Custom Data Sample

About the Attached Data Specification File

The attached data specification file (named **ccon_adata_spec.xml** by default) maps the key-value pairs (KVPs) in reporting event attributes to IDB tables and fields.

> **Important**
> For more information about attached data in Interaction Concentrator, see Processing Attached Data in the *Interaction Concentrator User's Guide*.

Uniqueness Requirements

When creating the attached data specification file, keep in mind the following requirement for unique key names.

In Vertical Tables

With respect to the vertical tables (that is, the public and secure `G_USERDATA_HISTORY` and `G_SECURE_USERDATA_HISTORY` tables), key name uniqueness is determined by a combination of three parameters:

- `<key_name>`
- `<source_id>` (userdata, reasons, extensions)
- `<storage_id>` (public, secure)

So a duplicate is a key name that is the same in all three parts. When it finds a duplicate key name,

ICON only uses the first occurrence of the key name definition and rejects the duplicates.

Vertical Tables Example 1:

```
<public>
<key name = "key1" source="userdata" history ="first"></key>
<key name = "key1" source="reasons" history ="first"></key>
</public>
```

These records are not duplicates because they have different <source_id> parameters.

Vertical Tables Example 2:

```
<public>
<key name = "key1" source="userdata" history ="first"></key>
</public>
<secure>
<key name = "key1" source="userdata" history ="first"></key>
</secure>
```

These records are not duplicates because they have different <storage_id> parameters.

In Horizontal Tables

The horizontal tables may include the following sections:

<call>, <call_cust>, <call_cust1>, <call_cust2>, <mcr-f>, <mcr-l>, <cust-disp-group-n>

For these sections key uniqueness depends on the following parameters:

- <storage_id> (<call>, <call_cust>, <call_cust1>, <call_cust2>, <mcr-f>, <mcr-l>, <cust-disp-group-n>)
- <field>

ICON only uses the first occurrence of the key definition and rejects duplicates. The key name parameter is not included when checking for uniqueness in horizontal tables because the same key name can be used multiple times if it is associated with different field or storage_id parameters.

Horizontal Tables Example 1:

```
<call_cust>
<key name = "key1" source="userdata" history ="first" field="cust-data-1"></key>
<key name = "key1" source="userdata" history ="first" field="cust-data-2"></key>
</call_cust>
```

These records are not duplicates because they have different <field> parameters.

Horizontal Tables Example 2:

```
<call_cust1>
<key name = "key1" source="userdata" history ="first" field="cust-data-1"></key>
</call_cust1>
<call_cust2>
<key name = "key1" source="userdata" history ="first" field="cust-data-1"></key>
</call_cust2>
```

These records are not duplicates because they have different <storage_id> parameters.

Schema Definition

The following is the XML schema definition for your attached data specification.

```
<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema targetNamespace="http://www.genesyslab.com/standards/icon/ed1"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:icon="http://www.genesyslab.com/standards/
icon/ed1" elementFormDefault="qualified" attributeFormDefault="unqualified">
<xsd:annotation>
  <xsd:documentation>Attached data configuration specification</xsd:documentation>
</xsd:annotation>

<xsd:simpleType name="AdataHistoryType">
<xsd:restriction base="xsd:string">
                  <xsd:enumeration value="none" />
<xsd:enumeration value="first" />
<xsd:enumeration value="last" />
<xsd:enumeration value="all" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="AdataCallHistoryType">
<xsd:restriction base="xsd:string">
<xsd:enumeration value="first" />
<xsd:enumeration value="last" />
        </xsd:restriction>
 </xsd:simpleType>

<xsd:simpleType name="AdataCallFieldType">
<xsd:restriction base="xsd:string">
                  <xsd:enumeration value="customer-segment" />
<xsd:enumeration value="service-type" />
                  <xsd:enumeration value="service-subtype" />
                  <xsd:enumeration value="busines-result" />
                  <xsd:enumeration value="customer-id" />
                  <xsd:enumeration value="transaction-id" />
                  <xsd:enumeration value="cause-id" />
                  <xsd:enumeration value="account-id" />
                  <xsd:enumeration value="destination-id" />
                  <xsd:enumeration value="target-id" />
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="AdataCallCustFieldType">
<xsd:restriction base="xsd:string">
                  <xsd:enumeration value="cust-data-1" />
                  <xsd:enumeration value="cust-data-2" />
                  <xsd:enumeration value="cust-data-3" />
                  <xsd:enumeration value="cust-data-4" />
                  <xsd:enumeration value="cust-data-5" />
                  <xsd:enumeration value="cust-data-6" />
                  <xsd:enumeration value="cust-data-7" />
                  <xsd:enumeration value="cust-data-8" />
                  <xsd:enumeration value="cust-data-8" />
                  <xsd:enumeration value="cust-data-10" />
                  <xsd:enumeration value="cust-data-11" />
                  <xsd:enumeration value="cust-data-12" />
                  <xsd:enumeration value="cust-data-13" />
                  <xsd:enumeration value="cust-data-14" />
                  <xsd:enumeration value="cust-data-15" />
                  <xsd:enumeration value="cust-data-16" />
                  <xsd:enumeration value="cust-data-17" />
```

```
                  <xsd:enumeration value="cust-data-18" />
                  <xsd:enumeration value="cust-data-19" />
  </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="AdataHistoryRecord">
          <xsd:attribute name="name" type="xsd:string" />
          <xsd:attribute name="source" type="icon:AdataSource" />
          <xsd:attribute name="history" type="icon:AdataHistoryType" />
</xsd:complexType>
<xsd:complexType name="AdataCallRecord">
          <xsd:attribute name="name" type="xsd:string" />
<xsd:attribute name="source" type="icon:AdataSource" />
          <xsd:attribute name="history" type="icon:AdataCallHistoryType" />
          <xsd:attribute name="field" type="icon:AdataCallFieldType" />
</xsd:complexType>
<xsd:complexType name="AdataCallCustRecord">
          <xsd:attribute name="name" type="xsd:string" />
          <xsd:attribute name="source" type="icon:AdataSource" />
          <xsd:attribute name="history" type="icon:AdataCallHistoryType" />
          <xsd:attribute name="field" type="icon:AdataCallCustFieldType" />
</xsd:complexType>
<xsd:element name="adata_spec">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="public" type="icon:AdataHistoryRecord" />
                  <xsd:element name="secure" type="icon:AdataHistoryRecord" />
                  <xsd:element name="call" type="icon:AdataCallRecord" />
                  <xsd:element name="call-cust" type="icon:AdataCallCustRecord" />
                  <xsd:element name="call-cust1" type="icon:AdataCallCustRecord" />
                  <xsd:element name="call-cust2" type="icon:AdataCallCustRecord" />
  </xsd:sequence>
  </xsd:complexType>
  </xsd:element>
  </xsd:schema>
```

Basic Sample

The following is an example of a basic **adata_spec.xml** file for voice and multimedia interactions. A similar sample file, **ccon_adata_spec_example.xml**, is available in your ICON installation folder after you install the application.

Attached data that you specify in the <public> and <secure> storage type sections of the specification file applies to all interactions. Attached data that you specify in the <call> and <call-cust> sections applies to voice calls only.

This sample includes examples of keys in a nested key-value list (in this case, attached data from Universal Routing Server [URS]). For more information about specifying URS attached data, see Universal Routing Server Attached Data.

```
<?xml version="1.0" encoding="utf-8" ?>
<adata_spec>
<public>
        <key name = "u_key1" source="userdata"    history ="first"/>
        <key name="u_key2" source="userdata" history="all" />
        <key name="RTargetAgentSelected" source="userdata" history="first" />
        <key name="RTargetUsed/RTargetName" source="userdata" history="first" />
        <key name="RTargetUsed/RTargetType" source="userdata" history="first" />
```

```
</public>

<secure>
        <key name="RTenant" source="userdata" history="first" />
</secure>

<call>
        <key name = "customer-segment" source="userdata" history ="first" field="customer-
segment"/>
        <key name = "svc_class_cd" source="userdata" history ="first" field="service-type"/>
        <key name = "CCTP_CALLTYPE" source="userdata" history ="first" field="service-
subtype"/>
        <key name = "cid" source="userdata" history ="first" field="customer-id"/>
        <key name = "transact_tn_final" source="userdata" history ="first" field="transaction-
id"/>
</call>

<call-cust>
        <key name = "STATE" source="userdata"  history ="last" field="cust-data-1"/>
</call-cust>

<call-cust1>
        <key name = "PegTD" source="userdata"  history ="last" field="cust-data-2"/>
</call-cust1>

<call-cust2>
        <key name = "vrapp_ctl_lang" source="userdata"  history ="last" field="cust-data-3"/>
</call-cust2>

</adata_spec>
```

Multimedia Sample

The following is an example of an **adata_spec.xml** file that has been modified to capture multimedia-specific attached data. A similar sample file, **ccon_adata_spec_mcr_example.xml**, is available in your ICON installation folder after you install the application.

With this attached data specification, values for the following predefined and customer-defined keys and attributes are stored in IDB in the GM_F_USERDATA or the GM_L_USERDATA table.

The GM_F_USERDATA table

The following user data key names and event attributes are always stored, whether or not they are defined explicitly in the XML file.

```
<mcr-f>
<key name = "FromPersonal" source="userdata"  history ="first" field="mcr-from-name"/>
<key name = "FromAddress" source="userdata"  history ="first" field="mcr-from-address"/>
<key name = "Subject" source="userdata"  history ="first" field="mcr-subject"/>
<key name = "Origination_Source" source="userdata"  history ="first" field="mcr-origin-
source"/>
<key name = "attr_itx_subtype" source="userdata" history="first"  field="mcr-sub-type”/>
<key name = "attr_itx_received_at" source="userdata" history="first"  field="mcr-received-
at”/>
</ mcr-f>
```

You can use the XML to define four additional keys that will be stored in the GM_F_USERDATA table. For example, you might define keys similar to the following ones:

```
<key name = "Custom Key Name - 1" source="userdata"  history ="first" field="mcr-reserved-1"/>
<key name = "Custom Key Name - 4" source="userdata"  history ="first" field="mcr-reserved-4"/>
```

The GM_L_USERDATA table

The following user data key names are always stored, whether or not they are defined explicitly in the XML file.

```
<key name = "SuggestedResponseID"         source="userdata"  history ="last" field="mcr-
suggested-response"/>
<key name = "AutoResponseID"                 source="userdata"  history ="last" field="mcr-
auto-response"/>
<key name = "AutoACKID"                 source="userdata"  history ="last" field="mcr-auto-
ack"/>
<key name = "Stop_Reason"                 source="userdata"  history ="last" field="mcr-stop-
reason"/>
<key name = "ContactId"                 source="userdata"  history ="last" field="mcr-ucs-
contact-id"/>
```

Predefined event attribute: attr_reason_system_name

Customer-defined keys: SuggestedResponseID, AutoResponseID, AutoACKID

For information about the IDB fields in which the values will be stored, see IDB Fields.

Sample adata_spec.xml File

In the GM_F_USERDATA section of the sample **adata_spec.xml** file that follows, the FromPersonal, Subject, and Origination_Source key names are defined, though they would appear in the table in any case. The CalledBack key name is user-defined and would not appear without the explicit definition. The event attributes are also pre-defined. In the GM_L_USERDATA table section, all of the key names are ones that are also predefined.

```
<?xml version="1.0" encoding="utf-8" ?>
<adata_spec>

<mcr-f>

     <key name = "FromPersonal" source="userdata"  history ="first" field="mcr-from-name"/>
     <key name = "FromAddress" source="userdata"  history ="first" field="mcr-from-address"/>
     <key name = "Subject" source="userdata"  history ="first" field="mcr-subject"/>
     <key name = "Origination_Source" source="userdata"  history ="first" field="mcr-origin-
source"/>
     <key name = "attr_itx_subtype" source="userdata" history="first"  field="mcr-sub-type”/>
    <key name = "attr_itx_received_at" source="userdata" history="first"  field="mcr-received-
at”/>

</ mcr-f>

<mcr-l>

     <key name = "SuggestedResponseID" source="userdata" history ="last" field=" mcr-
suggested-response "/>
     <key name = "AutoResponseID" source="userdata" history ="last" field=" mcr-auto-response
"/>
```

```
     <key name = "AutoAckID" source="userdata" history ="last" field=" mcr-auto-ack "/>
     <key name = " Stop_Reason " source="userdata" history ="last" field=" mcr-stop-reason "/>
     <key name = "ContactID" source="userdata" history ="last" field=" mcr-ucs-contact-id "/>

</ mcr-l>

</adata_spec>
```

Custom Data Sample

The following is an example of an **adata_spec.xml** file that has been modified for customized attached data processing.

You must create a custom dispatcher stored procedure to process the custom attached data. For an example of the script to create a custom dispatcher stored procedure, see Sample Script for Custom Attached Data.

```
<?xml version="1.0" encoding="utf-8" ?>

<adata_spec>

     <cust-disp-group-1>
        <key name = "name1_1" source="userdata" history ="first" field="cust-int-1"></key>
        <key name = "name2_1" source="userdata" history ="last" field="cust-int-2"></key>
        ...
        <key name = "name17_1" source="userdata" history ="last" field="cust-int-17"></key>
        <key name = "name18_1" source="userdata" history ="last" field="cust-str-1"></key>
        <key name = "name19-1" source="userdata" history ="last" field="cust-str-2"></key>
        ...
        <key name = "name34_1" source="userdata" history ="last" field="cust-str-17"></key>
     </cust-disp-group-1>

     <cust-disp-group-2>
        <key name = "name1_2" source="userdata" history ="first" field="cust-int-1"></key>
        <key name = "name2_2" source="userdata" history ="last" field="cust-int-2"></key>
        ...
        <key name = "name17_2" source="userdata" history ="last" field="cust-int-17"></key>
        <key name = "name18_2" source="userdata" history ="last" field="cust-str-1"></key>
        <key name = "name19_2" source="userdata" history ="last" field="cust-str-2"></key>
        ...
        <key name = "name34_2" source="userdata" history ="last" field="cust-str-17"></key>
     </cust-disp-group-2>

...

     <cust-disp-group-16>
        <key name = "name1_3" source="userdata" history ="first" field="cust-int-1"></key>
        <key name = "name2_3" source="userdata" history ="last" field="cust-int-2"></key>
        ...
        <key name = "name17_3" source="userdata" history ="last" field="cust-int-17"></key>
        <key name = "name18_3" source="userdata" history ="last" field="cust-str-1"></key>
        <key name = "name19_3" source="userdata" history ="last" field="cust-str-2"></key>
        ...
        <key name = "name34_3" source="userdata" history ="last" field="cust-str-17"></key>
     </cust-disp-group-16>

</adata_spec>
```

Sample Script for Custom Attached Data

This page presents a sample SQL script to create a custom dispatcher stored procedure—**gudCustDISP1** or **gudCustDISP2**—and a custom attached data storage table in your Interaction Database (IDB) schema.

The sample script in this appendix is for Microsoft SQL Server (MSSQL). After you install the Interaction Concentrator (ICON) application, sample scripts for each supported RDBMS type–**SampleProc_*db_type*.sql**–are available in the scripts subfolder in the directory to which you installed the application.

> ### Tip
> Carefully verify the syntax and operation of your modified **gudCustDISP1** or **gudCustDISP2** stored procedure. Any types of errors or RDBMS violations that the custom dispatcher stored procedure produces can affect ICON processing of all other attached data for voice calls and multimedia interactions.

The following MSSQL sample script illustrates how you can create a custom attached data storage table (G_SAMPLE_CUST_ADATA) and modify the **gudCustDISP1** or **gudCustDISP2** stored procedure in the **CoreProcedures_*db_type*.sql** script. The modified stored procedure stores arguments in the G_SAMPLE_CUST_ADATA table.

```
Sample Custom Dispatcher
/*==============================================================*/
/* Table: G_SAMPLE_CUST_ADATA                                   */
/*==============================================================*/
create table G_SAMPLE_CUST_ADATA (
   ID                   numeric(16)          identity,
   CALLID               varchar(50)          not null,
   CALL_TS              int                  not null,
   SWITCH_ID            int                  not null,
   TENANT_ID            int                  not null,
   C_INT_1              int                  null,
   C_INT_2              int                  null,
   ...
   C_INT_34             int                  null,
   C_STR_1              varchar(10)          null,
   C_STR_2              varchar(10)          null,
   ...
   C_STR_34             varchar(10)          null,
   GSYS_DOMAIN          int                  null,
   GSYS_PARTITION       int                  null,
   GSYS_SYS_ID          int                  null,
   GSYS_SEQ             bigint               null,
   GSYS_USEQ            bigint               null,
   GSYS_TS              datetime             null,
   GSYS_TC              int                  null,
   GSYS_EXT_VCH1        varchar(255)         null,
   GSYS_EXT_VCH2        varchar(255)         null,
   GSYS_EXT_INT1        int                  null,
   GSYS_EXT_INT2        int                  null,
```

```
   constraint PK_G_SAMPLE_CUST_ADATA primary key  (ID)
)
go

/*==============================================================*/
/* Index: IDX_G_SAMPLE_CDATA_TS                                 */
/*==============================================================*/
create   index IDX_G_SAMPLE_CDATA_TS on G_SAMPLE_CUST_ADATA (
CALL_TS ASC
)
go

-- ============================================================ --
-- Name:   -- gudCustDISP1 --
-- Group:  User data related procedures
-- Brief:  -- gudCustDISP DISPATCH --
-- ============================================================ --
DROP PROCEDURE gudCustDISP1
go

CREATE PROCEDURE gudCustDISP1
        @GROUPID                INTEGER,
        @CALLID                 VARCHAR(64),
        @P_CALL_TS              INTEGER,
        @SWITCHID               INTEGER,
        @TENANTID               INTEGER,
        @TS_S                   INTEGER,
        @TS_MS                  INTEGER,
        @P_STR_1                VARCHAR(255),
        @P_STR_2                VARCHAR(255),
        ...
        @P_STR_17               VARCHAR(255),
        @P_INT_1                INTEGER,
        @P_INT_2                INTEGER,
        ...
        @P_INT_17               INTEGER,
        @P_STR_RES1             VARCHAR(255),
        @P_STR_RES2             VARCHAR(255),
        @P_STR_RES3             VARCHAR(64),
        @P_STR_RES4             VARCHAR(255),
        @P_INT_RES1             INTEGER,
        ...
        @P_INT_RES7             INTEGER,
        @DOMAIN                 INTEGER,
        @PARTITION              INTEGER,
        @SYS_ID                 INTEGER,
        @SYS_SEQN               NUMERIC(20,0),
        @SYS_TS                 DATETIME
AS
BEGIN

    -- Insert first portion in the long table --
    IF (@GROUPID = 1)
    BEGIN
        INSERT INTO G_SAMPLE_CUST_ADATA(
                CALLID,
                CALL_TS,
                SWITCH_ID,
                TENANT_ID,
                C_INT_1,
                C_INT_2,
                ...
                C_INT_17,
```

```
                C_STR_1,
                C_STR_2,
                ...
                C_STR_17,
                GSYS_DOMAIN,
                GSYS_PARTITION,
                GSYS_SYS_ID,
                GSYS_SEQ,
                GSYS_USEQ,
                GSYS_TS)
        VALUES (
                @CALLID,
                @P_CALL_TS,
                @SWITCHID,
                @TENANTID,
                @P_INT_1,
                @P_INT_2,
                ...
                @P_INT_17,
                @P_STR_1,
                @P_STR_2,
                ...
                @P_STR_17,
                @DOMAIN,
                @PARTITION,
                @SYS_ID,
                @SYS_SEQN,
                0,
                @SYS_TS)

    END

    -- Update record and specify more fields --
    ELSE
    IF (@GROUPID = 2)
    BEGIN
        UPDATE G_SAMPLE_CUST_ADATA SET
                C_INT_18                = @P_INT_1,
                ...
                C_INT_34                = @P_INT_17,
                C_STR_18                = @P_STR_1,
                C_STR_19                = @P_STR_2,
                ...
                C_STR_34                = @P_STR_17,
                GSYS_USEQ               = @SYS_SEQN
        WHERE CALLID = @CALLID AND CALL_TS = @P_CALL_TS


    END
END
go
```

Appendix: Log Events

The following log events have been added or have been updated since the most recent update to the *Combined Log Events Help*:

09-20034	09-20035	09-20036	09-20037	09-20038	09-20039	09-20040	09-25024	09-25025	09-25026
09-25030	09-25031	09-25032	09-25033	09-25034	09-25035	09-25036	09-25100	09-25102	
09-25103	09-25104	09-25105	09-25106	09-25107	09-25108	09-25109			

Refer to the *Combined Log Events Help* for information about the following log events:

09-20008	09-20009	09-20010	09-20018	09-20020	09-20021	09-20022	09-20023	09-20024
09-20025	09-20026	09-20027	09-20028	09-20029	09-20030	09-20031	09-20032	09-20033
09-25001	09-25002	09-25003	09-25004	09-25005	09-25006	09-25007	09-25008	09-25009
09-25010	09-25011	09-25012	09-25013	09-25014	09-25015	09-25016	09-25017	09-25018
09-25019	09-25020	09-25021	09-25022	09-25023	09-25027	09-25028	09-25029	09-25030
09-25031								

09-20034

Level	Standard
Text	On Object Destroyed: ObjectType=[*object_type*], DBID=[*object_dbid*], Name=[*object_name*]
Attributes	[*object_type*]—Configuration Server Object Type [*object_dbid*]—Configuration Server Object DBID [*object_name*]—Configuration Server Object Name
Description	Indicates that a configuration object has been removed from ICON's internal storage.
Actions	No action is required.

09-20035

Level	Trace
Text	Request TRegisterAddress for [*DN_name*] failed.
Attributes	[*DN_name*]—The name of the DN in the Configuration Database.
Description	Enables you to monitor DN registration. EventError was received on request TRegisterAddress().
Actions	Check for the presence of the ThisDN attribute.

09-20036

Level	Trace
Text	All devices are successfully registered on link [Server='[*server_name*]'; Switch='[*switch_name*]'].
Attributes	'[*server_name*]'—The name of the T-Server or Interaction Server serving the switch on which the DNs are being registered. '[*switch_name*]'—The name of the switch on which the DNs are being registered. Together these uniquely identify the link for which the DNs/endpoints are registered.
Description	ICON has read the configuration, connected to the specified Server, and completed registration of all DNs/endpoints.
Actions	No action is required.

09-20037

Level	Trace
Text	All devices on all CTI links registered.
Attributes	No attributes.
Description	ICON has read the configuration, connected to all of the T-Server/SIP Server instances listed on the Interaction Concentrator Application Connections tab, and completed registration of all DNs associated with those servers.
Actions	No action is required.

09-20038

Level	Trace
Text	All devices on all MCR links registered.
Attributes	No attributes.
Description	ICON has read the configuration, connected to all of the Interaction Server instances listed on the Interaction Concentrator Application Connections tab, and completed registration of all endpoints associated with those Interaction Servers.
Actions	No action is required.

09-20039

Level	Standard

Text	Call-processing error occurred on the link [T-Server='[*t-server_name*]'; Switch='[*switch_name*]'], CallUUID ['[*calluuid*]';'[*timestamp*]']. Check application logs.
Attributes	'[*t-server_name*]'—The name of the T-Server `Application` object '[*switch_name*]'—The name of the `Switch` object '[*calluuid*]'—The CallUUID '[*timestamp*]'—The timestamp indicating when the call was created
Description	If any of the errors that can trigger this alarm occur and you have configured the log-call-failure option to log such errors, ICON prints a Standard-level error message noting the call-processing problem. For details, see Setting Alarms for Call Processing Failures in the *Interaction Concentrator User's Guide*.
Alarm Advisory	Set an alarm to monitor whether an environmental condition is causing call failures that are resulting in data loss.
Actions	Check your environment for conditions that might be causing a high number of call failures.

09-20040

Level	Standard
Text	'[*message_text*]'. Check configuration.
Attributes	'[*message_text*]' - The message can be one of the following: • Agent not found by Agent Login ID='[*login_id*]' on the link Switch='[*switch_name*]' • Agent Login not found by Agent Login ID= '[*login_id*]' on the link Switch='[*switch_name*]' • Person having Employee ID='[*employee_id*]' not found in tenant '[*tenant_name*]'
Description	Indicates that a required Agent Login or Person configuration object was not found in your environment.
Alarm Advisory	Set this alarm to warn you about possible inconsistencies in your environment. There is no clearance event.
Actions	Check that specified configuration object exists in your environment.

09-25024

Level	Standard
Text	ICON cannot preserve or store the data: [*errtxt*].
Attributes	'[*errtxt*]'—Specific details regarding the error
Description	Interaction Concentrator cannot preserve or store the data. There are many events that could cause ICON to generate this error, including operating systems errors, disk full, insufficient memory, system queue corruption, or internal software errors. Manual intervention is required.
Alarm Advisory	Set an alarm to monitor whether problems with internal PQ file structure are causing ICON to stop processing data, resulting in data loss.
Actions	1. Check for and correct any operating system errors. 2. Check that your disk is not full. 3. If either step does not resolve the problem, contact Genesys Technical Support for investigation of possible internal error.

09-25025

Level	Standard
Text	Persistent storage backlog threshold reached: [*errtxt*].
Attributes	'[*errtxt*]'—Specific details regarding the backlog
Description	Using the pq-backlog-alarm-threshold configuration option, you have defined a threshold that ICON is to observe when it is unable to write records to IDB. If this event appears, this limit has been reached and ICON can no longer backlog records to memory. It is common for ICON to log this event following 09-25024.
Alarm Advisory	Indicates abnormal conditions. You might consider setting an Alarm Condition for this event. The correct Cancel Event is 09-25026.
Actions	One of the configured backlog thresholds of persistent storage has been reached. 1. Check that threshold parameter is properly set. 2. Check the ICON and related T-Server logs for errors.

	3. Correct any problems that you discover. The amount of time it takes to clear the backlog depends on how much data from the abnormal condition must be flushed.

09-25026

Level	Standard
Text	Persistent storage backlog submitted: [*description*].
Attributes	'[*description*]'—Specific details of the backlog that Interaction Concentrator has flushed to the Interaction database.
Description	An informational message indicating that Interaction Concentrator has flushed the persistent storage backlog to the Interaction Database and now operates under normal conditions. The sequence of events tripped by the 09-25025 log event no longer presents an alarming condition.
Alarm Advisory	This is the clearance event for 09-25025.
Actions	None.

09-25030

Level	Standard
Text	All persistent queues for [*pq_manager*] connected to database.
Attributes	[*pq_manager*]—The name of the persistent queue manager.
Description	Reports that all persistent queue connections for the specified manager have been connected.
Alarm Advisory	Clearance event for 09-25031.
Actions	Use Genesys Administrator or GAX to clear the alarm you set for 09-25031.

09-25031

Level	Standard
Text	Persistent queue [*pq_name*] for [*pq_manager*] disconnected from database.
Attributes	[*pq_name*]—The name of the disconnected persistent queue. [*pq_manager*]—The name of the persistent queue manager.
Description	Indicates that the specified persistent queue has

	become disconnected.
Alarm Advisory	Use this event to trigger an alarm in Genesys Administrator or GAX. 09-25030 is the clearance event for this alarm.
Actions	Check the availability of DBServer and IDB and the connections to DBServer and IDB. **Warning** The connection must be restored within 24 hours to prevent loss of data, because ICON has a limited capacity to store reporting data locally.

09-25032

Level	Standard
Text	DAP [*dap_name*] – [*pq_mgr_name*], Queue [*queue_name*]: Encoding is non-unicode, [*encoding_type*].
Attributes	[*dap_name*]—DAP name [*pq_mgr_name*]—Persistent Queue manager name [*queue_name*]—Database queue identifier [*encoding_type*]—The current database encoding type
Description	The support-unicode option is not configured, or it has been set to the value of `0`, or the IDB encoding is non-Unicode, or for Microsoft SQL IDBs the DB Client version does not support Unicode. **Important** In the case of Microsoft SQL IDBs, the text might specify 'nvarchar' instead of 'varchar'.
Alarm Advisory	If you plan to use Unicode encoding, set an alarm for this message. 09-25033 is the clearance event for this alarm.
Actions	If you are using a Unicode IDB, configure the **support-unicode** option using the value 1 and verify that the IDB encoding is set for Unicode. If you are using a Microsoft SQL IDB, upgrade your DB Client version to one that supports Unicode. For more information, see Configuring for Unicode Support in an Environment with a Microsoft SQL IDB.

09-25033

Level	Standard
Text	DAP [*dap_name*] – [*pq_mgr_name*], Queue [*queue_name*]: Unicode encoding [*encoding_type*] is configured and supported.

Attributes	[*dap_name*]—DAP name [*pq_mgr_name*]—Persistent Queue manager name [*queue_name*]—Database queue identifier [*encoding_type*]—The actual encoding, such as UTF8
Description,	Indicates that ICON has checked that the IDB encoding is Unicode, that the support-unicode option is set to the value 1 and, for Microsoft SQL IDBs, that the DB Client version supports Unicode.
Alarm Advisory	This is the clearance event for 09-25032.
Actions	No action is necessary.

09-25034

Level	Standard
Text	DAP [*dap_name*] – [*pq_mgr_name*], Queue [*queue_name*]: ICON is incompatible with Configuration Server encoding.
Attributes	[*dap_name*]—DAP name [*pq_mgr_name*]—Persistent Queue manager name [*queue_name*]—Database queue identifier
Description	Indicates that the support-unicode option is either not configured or has been set to the value of `0`; ICON also checks support of Unicode by Configuration Server.
Alarm Advisory	Set this alarm to warn you about possible inconsistencies in your environment. The clearance event is 09-25035.
Actions	Resolve this inconsistency according to what your environment requires—Unicode or non-Unicode encoding—and then restart ICON.

09-25035

Level	Standard
Text	DAP [*dap_name*] – [*pq_mgr_name*], Queue [*queue_name*]: ICON is compatible with Configuration Server encoding.
Attributes	[*dap_name*]—DAP name [*pq_mgr_name*]—Persistent Queue manager name [*queue_name*]—Database queue identifier
Description	Confirms that your ICON encoding is compatible with the Configuration Server encoding.
Alarm Advisory	This is the clearance event for 09-25034.
Actions	No action is necessary.

09-25036

Level	Standard
Text	This log event number can have either of two messages: • DAP [*dap_name*] – [*pq_mgr_name*], Queue [*queue_name*]: IDB encoding does not support Unicode. or • DAP [*dap_name*] – [*pq_mgr_name*], Queue [*queue_name*]: DBClient version is [*dbclient_version*]. Version [*dbclient_required_version*] or higher is required to support Unicode.
Attributes	[*dap_name*]—DAP name [*pq_mgr_name*]—Persistent Queue manager name [*queue_name*]—Database queue identifier [*dbclient_version*]—Your currently-installed version of DB Server's DB Client [*dbclient_required_version*]—The earliest release of the DB Server DB Client that supports current Unicode functionality
Description	ICON checks your IDB encoding. If the encoding doesn't support Unicode, then ICON prints the first event message listed above and shuts down. If the IDB encoding does support Unicode, then, *for Microsoft SQL IDBs only*, ICON checks the DB Client version. If the DB Client version does not support Unicode, ICON prints the second of the two event messages above and shuts down. For more information about DB Client, see the DB Server User's Guide.
Alarm Advisory	Set an alarm to warn you of configuration inconsistencies that can prevent ICON from running.
Actions	Change your IDB encoding and, if necessary, upgrade your DB Client version.

09-25100

Level	Standard
Text	Adata: Parsing error – [*errtxt*] [*context*].
Attributes	[*errtxt*]—The text of the error message. [*context*]—The section of the attached data specification file that contains the error.
Description	Interaction Concentrator has encountered an error while parsing the attached data specification XML

	file.
Alarm Advisory	Indicates abnormal conditions. You might consider setting an Alarm Condition for this event.
Actions	The attached data specification XML file should be modified according to the User Documentation to eliminate the cause of the error.

09-25102

Level	Trace
Text	Adata: Loading data specification from: [*source*].
Attributes	[*source*]—The name of the attached data specification file.
Description	The attached data specification is being loaded from the XML file. For more information, see Attached Data Specification File.
Actions	No action is required.

09-25103

Level	Trace
Text	Adata: New dictionary created from [*source*], id = [*dictionary_id*], num = [*dictionary_number*].
Attributes	[*source*]—The name of the new attached data source file. [*dictionary_id*]—The ID given to the new attached data dictionary. [*dictionary_number*]—The number of attached data dictionaries that currently exist.
Description	The attached data specification file was successfully loaded and the new key specifications have been applied by ICON. A new dictionary of key names has been created. For a more detailed explanation of the attached data dictionary, see Processing Attached Data in the *Interaction Concentrator User's Guide*.
Actions	No action is required.

09-25104

Level	Trace
Text	Adata: Key registered: [*storage*], [*keyname*], [*data_source*], [*history_type*], [*field_name*].
Attributes	[*storage*]—Storage type, such as `public`, `secure`, `call`, `custom`, and so on.

	[*keyname*]—The name of the key. [*data_source*]—Which event attribute ICON uses to extract the key name, such as `userdata`, `reasons`, or `extensions`. [*history_type*]—The keyname history type, such as `last`, `first`, `all` or `none`. [*field_name*]—The attribute `field` value or an empty string.
Description	Interaction Concentrator has registered the attached data key from the attached data specification file.
Actions	No action is required.

09-25105

Level	Standard
Text	Adata: Duplicate of user key name, ignored: new [*storage1*], [*keyname1*], [*data_source1*], [*history_type1*], [*field_name1*]; Old [*storage2*], [*keyname2*], [*data_source2*], [*history_type2*], [*field_name2*].
Attributes	**The attribute definitions are the same for the old and new keynames.** [*storage*]—Storage type, such as `public`, `secure`, `call`, `custom`, and so on. [*keyname*]—The name of the key. [*data_source*]—Which event attribute ICON uses to extract the key name, such as `userdata`, `reasons`, or `extensions`. [*history_type*]—The keyname history type, such as `last`, `first`, `all` or `none`. [*field_name*]—The attribute `field` value or an empty string.
Description	A key duplication error has occurred while parsing the attached data specification file. The attribute definitions for the new keyname are the same as for the existing (old) keyname.
Actions	Edit the attached data specification file to eliminate the duplicate key.

09-25106

Level	Standard
Text	Adata: The value of [*key_name*] has been truncated because the length exceeded the database field size.
Attributes	[*key_name*]—The name of the attached data key for which the value has been truncated in order to comply with length limits for key values.

Description	The maximum size is 255 characters.
Actions	Edit the attached data key values so that they contain 255 characters or fewer.

09-25107

Level	Standard
Text	Adata: Key name [*source_key_name*] will be truncated to [*truncated_key_name*].
Attributes	[*source_key_name*]—The key name as you provided in the attached data specification file. [*truncated_key_name*]—The key name as it has been truncated in order to comply with the length limits for key names.
Description	The key name will be truncated because the length exceeds the database field size. This message appears only for vertical tables; that is, the the public and secure G_USERDATA_HISTORY and G_SECURE_USERDATA_HISTORY tables.
Actions	No action is necessary. However, you might choose to edit the key name so that it will not be truncated.

09-25108

Level	Standard
Text	Adata: Dictionary from [*source_file_name*] was not created. Reason: [*reason_text*].
Attributes	[*source_file_name*]—Name of the new dictionary you are trying to create. [*reason_text*]—Describes the problem with the attached data specification file.
Description	Possible reasons include the following: the maximum number of simultaneously existing dictionaries in memory was reached (by default this is twelve); or, the dictionary is actually a duplicate of the previous one.
Actions	Wait until at least one dictionary is released (that is, all associated calls have been ended) or eliminate duplicate records from the attached data specification file.

09-25109

Level	Standard
Text	The value of [*configuration_object_name*] attribute

	[*attribute_name*] has been truncated to [*truncated_attribute_value*] because the length exceeded the database field size.
Attributes	[*configuration_object_name*]—The name of the configuration object with which the specified attribute is associated. [*attribute_name*]—The name of the attribute that has been truncated. [*truncated_attribute_value*]—The new, shortened value created by truncating the original value.
Description	Applies to Virtual Agent Group script expressions. If the expression exceeds 255 characters, ICON truncates it to make it fit the designated database field size.
Actions	You can use the truncated version of the attribute value or edit the value to abide by the limit.

Appendix: Migration Procedures

> **Important**
> This topic contains migration information only for release 8.1.5, and higher. For instruction on how to migrate to earlier releases of Interaction Concentrator, see the Interaction Concentrator chapters in the Genesys Migration Guide.

Migration Preliminaries

Complete the following procedures before starting your migration of Interaction Concentrator:

1. Migrate Management Framework, as applicable for your deployment.
2. Upgrade other prerequisite Genesys components (for example, T-Server, Interaction Server, or Universal Routing Server), as applicable for your deployment.
3. Update the contact center configuration (for example, Place Groups, Agent Groups, and DNs).

Migration Procedures

Main Steps

1. Migrate your ICON Server.
2. Check whether you need to update your IDB schema.
3. If necessary, migrate your IDB schema.

In a High Availability Environment

Genesys recommends that you run both ICONs in each ICON pair simultaneously for at least one day before the migration. At the minimum, the time the ICONs run simultaneously must exceed the time set in the Genesys Info Mart **max-call-duration** configuration option.

The recommended upgrade plan is the following:

1. Stop one member of an ICON pair, migrate that ICON instance, and then restart it. Make sure it starts successfully. If not, roll back the migration (see Rolling Back the Migration).
2. If the ICON migration was successful:
   - For voice or SIP chat interactions: Wait until the longest active interaction in the environment finishes (usually 24 hours is enough).
   - For multimedia (e-mail, non-SIP chat, other third-party media) interactions: To avoid data quality

issues, Genesys recommends that you wait for a period (usually 24 hours is enough).

3. Then—assuming the migrated ICON has been operating normally—stop the other member of the HA pair of ICONs and perform the same migration procedure, again ensuring that the migration is successful.

In an Environment with Genesys Info Mart

> **Important**
>
> Your migration should be done in a working environment with an available ICON database and the associated ICON DBServer.

If Genesys Info Mart is running in your environment, stop all Genesys Info Mart extraction jobs before you start your ICON migration. Restart Job_ExtractICON only after you have successfully completed the migration of the ICON instances from which Genesys Info Mart is extracting data.

If you have multiple ICON applications, repeat the migration procedure for each ICON instance (including each ICON in any HA pair). You do not have to upgrade all ICON instances at the same time; Genesys Info Mart can operate for a period with different versions of ICON (even within the same HA pair). However, be aware that you risk inconsistent data processing if you run different versions of ICON at the same time.

Migrating the ICON Server

Follow these steps to migrate the server component of Interaction Concentrator to the latest release:

1. Find a timeframe for this migration during which ICON has minimal activity.
2. Stop ICON, either using Genesys Solution Control Interface (SCI), Genesys Administrator/Genesys Administrator Extension, or manually. This may take several minutes. Make sure not to restart ICON until after the database update is complete.
3. Back up the Interaction database (IDB).
4. If you have customized the gudCustDISP1 or gudCustDISP2 dispatcher stored procedures and saved them with the default (Genesys-provided) names, back up the procedures and restore them after the upgrade.
5. Back up the entire ICON directory in case you need to roll back the migration.
   - Back up your customer-related files, which have the following suffixes:
     - ***.db**
     - ***.pq**
     - ***.xml**—You can locate this file, which by default is named **ccon_adata_spec.xml**, in the directory where the current version of Interaction Concentrator is installed.
   - Save the scripts that drop outdated stored procedures. To do so, save the **drop_<version>_<db>.sql** file to be used after successful migration completion to drop the old

stored procedures. Keep the old stored procedures for approximately one week. Then you can drop them, if you prefer, but there is no need to.

6. Run the setup file (**install.sh** or **setup.exe**) provided in your installation package.

> **Important**
>
> These instructions assume you are installing the new version of ICON in the same location as the old one. If you install the upgraded version into a different directory from the existing version, verify that you are launching the correct ICON version when you restart. For example, you might need to adjust the Windows Service settings. For instructions on how to configure ICON start up, see Starting and Stopping.

7. Restore the backup customer-related files saved in Step 5.
   - Replace the files installed in the folder where you deployed ICON 8.1.x during migration with your saved files.
8. For environments running any Genesys Info Mart release prior to 8.5.007.xx, run the **update_idb_for_gim.sql** or **update_idb_for_gim_mm.sql** script, as appropriate, every time you migrate to a new release of Interaction Concentrator.

Checking Your IDB Schema Version

Check if an ICON database update is necessary by comparing database schema versions.

1. The database schema version of the new installation can be identified from the name of the following file: **drop_<*VERSION*>_<*db_type*>.sql**
   where
   - *<VERSION>* stands for the new database version. The version has the format *N.N.NNN.NN*.
   - *<db_type>* stands for the short name of the database type.

   For example, the filename might be: **drop_8.1.512.xx_postgre.sql**.
2. The currently-installed RDBMS version can be identified by either the name of the **drop_<*VERSION*>_<*db_type*>.sql** file in the previous installation of ICON or using the following database query:
   ```
   SELECT VAL FROM G_DB_PARAMETERS where SECT = 'schema' and OPT = 'version'
   ```

> **Important**
>
> If both the current and new ICON schema versions are the same, you do not need to update your IDB. The version of the database schema does not necessarily match the release number of the component. For example, for Interaction Concentrator release 8.1.509.09, the database schema version is 8.1.508.05.

Updating IDB

The update includes an upgrade of the stored procedures and it might include a change to the database schema.

To perform the schema upgrade, apply all the upgrade scripts sequentially in the order of increasing versions starting from the lowest numbered upgrade script version above the current (old) ICON database schema version through the highest version found in the directory.

The Interaction Concentrator installation package may contains one or more upgrade schema scripts in the following format: **Upgrade_N.N.NNN.NN_<db_type>.sql**
where:

- *N.N.NNN.NN* indicates the schema version to which this script upgrades the database schema.
- *<db_type>* defines the type of database (`ora`, `mssql`, `db2`, or `postgre`).

Examples

Example A

If the old IDB version is 8.1.100.28 on ORACLE and the new ICON installation provides the following upgrade scripts:

- **Upgrade_8.1.100.27_ora.sql**
- **Upgrade_8.1.100.29_ora.sql**
- **Upgrade_8.1.100.32_ora.sql**

Then run the following scripts in the specified order:

- **Upgrade_8.1.100.29_ora.sql**
- **Upgrade_8.1.100.32_ora.sql**

Example B

If the old IDB version is 8.1.100.32 on ORACLE and the new ICON installation provides the following upgrade scripts:

- **Upgrade_8.1.100.27_ora.sql**
- **Upgrade_8.1.100.29_ora.sql**
- **Upgrade_8.1.100.32_ora.sql**

There is no need to apply any of the upgrade scripts.

Example C

If the old IDB version is 8.1.100.35 on ORACLE and the new ICON installation provides the following upgrade scripts:

- **Upgrade_8.1.100.27_ora.sql**
- **Upgrade_8.1.100.29_ora.sql**
- **Upgrade_8.1.100.32_ora.sql**

There is no need to apply any of the upgrade scripts.

Example D

If the old IDB version is 8.1.100.27 on ORACLE and the new ICON installation provides the following upgrade scripts:

- **Upgrade_8.1.100.27_ora.sql**
- **Upgrade_8.1.100.29_ora.sql**
- **Upgrade_8.1.100.32_ora.sql**

Then run the following scripts in the specified order:

- **Upgrade_8.1.100.29_ora.sql**
- **Upgrade_8.1.100.32_ora.sql**

Upgrading the Stored Procedures

1. To create a new set of stored procedures in IDB to support your new ICON installation, run **CoreProcedures_<*db_type*>.sql**.
2. If you are running Oracle 11 or higher and plan to purge by truncating partitions, you must create a new partitioned IDB. See Configuring a Partitioned Oracle IDB in the *Interaction Concentrator Deployment Guide* for deployment considerations and instructions.

   > **Important**
   > If you have an existing partitioned IDB, you cannot migrate that IDB to a non-partitioned database structure.

3. Upgrade your purge procedure. To do so, execute the appropriate one of the following scripts:
   - **Purge2_<*database*>.sql**
   - **PurgePart_ora.sql**—If you have a partitioned IDB.
   - **Purge2_PartitionType0_ora.sql**—If you have a non-partitioned Oracle IDB with the ICON **partition-type** configuration option set to `0`. Available in release 8.1.505.05 and higher.
4. Verify the purge parameters. Verify that the GSYSPurge81 procedure has been set up appropriately for your deployment. To do this:
   a. Retrieve the **transaction_size** value by selecting
   ```
   SELECT Val
   from
   G_DB_PARAMETERS
   where
   SECT = 'GSYSPurge81'
   ```

```
and
OPT = 'rowspertransaction'
```

b. Make sure `Val` has the required value (the default value is `200,000` records). If necessary, you can make changes manually or you can use an optional Interaction Concentrator stored procedure, svcUpdateDBParameters, provided in the **Wrapper_for_<*version*>_<*db_type*>.sql** script.

5. Upgrade the wrappers. The ICON scripts directory contains wrappers for the stored procedures. For each RDBMS type, there is a file that matches pattern **Wrapper_for_*_<*db_type*>.sql**. For example, for Microsoft SQL, the wrapper name is **Wrapper_for_<*version*>_mssql.sql**.
   - Execute the wrapper for your RDBMS.

Re-Starting ICON and Verifying the Migration

1. Start the upgraded ICON. Genesys recommends that you use Solution Control Interface (SCI) or Genesys Administrator/Genesys Administrator Extension to start ICON.
2. Wait while ICON completes startup operations. The startup time strictly depends on the size of your configuration environment and may take more than 30 minutes.
3. Check the log for errors to verify that ICON is running properly. The criteria for a successful migration is that ICON starts and no database problem is detected.
   ICON is started when all three of the following trace messages below have been printed in the log:
   - Trace log message 09-25004: `Database queue [ID]: persistent queue transaction [number1] is processed, committed and removed. [number2] records are written.`
   - Trace log message 09-25016: `Persistent Queue [ID]: transaction [number1] is committed. [number2] records written into the queue.`
   - Trace log message 00-04541: `Message [message type] received from [socket number] ([application type] [application name]).`

   If ICON is unable to write interaction data in IDB, then see Rolling Back the Migration for instruction on reversing the migration and restoring your previous version of ICON.

Follow-Up Steps to Complete Your Migration

1. If ICON started successfully, you can start the Genesys Info Mart Job_ExtractICON.
2. (Optional) After a week—to allow time to be certain that you will not need to roll back your upgrade—you can delete all old stored procedures for previous Interaction Concentrator releases. To do so, execute the **drop_<*previous_schema_version*>_<*db_type*>.sql** script you saved in Step 5 in "Migrating the ICON Server" (above).

Rolling Back the Migration

If the upgrade failed and ICON did not start successfully, perform the following steps:

1. Stop Interaction Concentrator.
2. Restore the IDB database from your backup.
3. Restore entire the Interaction Concentrator directory, including the **icon.exe** file, as well as the ***.pq** and ***.db** files.
4. Start the restored version of ICON.
5. If ICON starts successfully, you can then start the Genesys Info Mart extraction job.

Changes to the Migration Procedures in Release 8.1.502.04

Starting in release 8.1.502.04, you can add applications of the Interaction Server type to the ICON application `Connections` tab. In earlier releases, connecting to Interaction Server required you to use an application of the T-Server type that was configured to contain the connection parameters for Interaction Server.

- If you have connections to Interaction Server that were created using a specially-configured T-Server application, you can continue using them. You do not need to change your existing connections to Interaction Server.
- If you create a new connection to Interaction Server, use the Interaction Server application type.

Important Migration Notes

- You can migrate Interaction Concentrator from any version of release 7.5 or 7.6 directly to release 8.x.
- To migrate Interaction Concentrator 7.2 to 8.x, first, migrate to release 7.5 or 7.6 and then migrate to release 8.x.
- ICON processing must be suspended while the Interaction Concentrator migration is occurring. The content of the persistent queue file (**icon_*.pq**) is lost as a result of the upgrade. To minimize the loss of data, perform the upgrade when the contact center load is minimal.
- Stopping ICON does not stop the execution of any stored procedures that may be running or scheduled to run during the upgrade. In particular, if the merge or purge stored procedures are executing when you run the database scripts, these stored procedures are not successfully dropped and then re-created. You are not notified if you did not upgrade these service procedures correctly.
- You cannot migrate from a non-partitioned IDB to a partitioned IDB, or vice versa. If you want to start or stop using partitions, you must create a new IDB. (Partitioning is available on Oracle RDBMSs only.) For detailed information on this topic, see Purging by Truncating Partitions in the *Interaction Concentrator User's Guide*.