SIP Cluster Solution Guide

SIP Cluster Current

12/30/2021

Table of Contents

SIP Cluster Solution Guide

> **Important**
> SIP Cluster solution is under restricted availability. Contact Product Management for more information.

Use this guide to find a general overview of how the SIP Cluster works, as well as all special configuration of Genesys components related to the SIP Cluster solution.

About SIP Cluster

Find out how SIP Cluster components and applications work.

- Overview
- Cluster components
- Unsupported functionality
- Deploying SIP Cluster

Component Configuration

Find procedures to set up the core components.

- Configuring SIP Server
- Configuring SIP Proxy
- Configuring SIP Feature Server
- Stat Server

Routing Principles

Find out about a routing concept in SIP Cluster.

- Routing principles
- Agent availability for routing
- Agent reservation
- Routing optimization
- Routing limitations

Historical Reporting

Find out about historical reporting in SIP Cluster.

- Overview
- Architecture
- Deployment considerations
- Operational considerations
- Deployment tasks
- SIP Business Continuity

About SIP Cluster

> **Important**
>
> SIP Cluster solution is under restricted availability. Contact Product Management for more information.

The SIP Cluster solution provides maintenance simplicity for the large Genesys deployments by treating multiple SIP Servers as a single system across multiple locations and by distributing call and Agent/DN state processing among SIP Cluster nodes. The cluster is geographically aware and can maintain the geographical integrity of calls. Once a call is assigned to a SIP Server node, it is maintained exclusively by that node.

When running in a cluster mode, SIP Server processes calls independently from Agent/DN state processing, meaning that a call might be handled on one SIP Server instance while an Agent/DN state is maintained on another SIP Server instance.

The SIP Proxy layer—between Session Border Controllers (SBC) and SIP Servers—load balances an inbound traffic by distributing calls to available SIP Server nodes in a data center. Agent/DN state ownership by a SIP Cluster node enables each node to maintain a subset of Agent/DN states.

Key Features

- **Configuration simplicity**
  SIP Cluster one-switch architecture simplifies maintenance procedures as well as it allows adding new SIP Server applications to the cluster with minimal configuration changes.
- **Architecture transparent to clients**
  Devices connect and register to the SIP Cluster (through SIP Proxy), instead of to a particular node in the network.
- **Business Continuity**
  Build a more robust network, with work area redundancy, provision a solution with build-in business continuity attributes, and graceful migration.
- **Solution integration**
  SIP Cluster integrates with most other Genesys systems and solutions, including:
  - Historical Reporting (Interaction Concentrator, Genesys Info Mart, and Genesys Interactive Insights)
  - Genesys Voice Platform for IVR and media treatments
  - Genesys Interaction Recording
  - Outbound Contact Server
  - Digital Solutions (Chat, Email, SMS, etc.)

More about SIP Cluster

- SIP Cluster components and their versions
- Deployment steps

SIP Cluster Components

> **Important**
> SIP Cluster solution is under restricted availability. Contact Product Management for more information.

The SIP Cluster includes several core and supporting components:

- SIP Server provides the hardware-to-software interface for calls.
- SIP Proxy acts as SIP registrar and hides the SIP Server infrastructure from SIP endpoints.
- SIP Feature Server handles voicemail, provides the user interface for configuring and using dial plans.
- Orchestration Server and Universal Routing Server provide voice-routing capabilities.
- Stat Server provides support for the reporting of interactions in the cluster environment.
- Historical Reporting in the SIP Cluster solution provides detailed data, analytics, and reports about activity in your entire contact center, through use of several Genesys products:
  - Interaction Concentrator
  - Genesys Info Mart
  - Reporting and Analytics Aggregates (RAA)
  - Genesys Customer Experience Insights (GCXI) (which replaces the deprecated Genesys Interactive Insights [GI2])
- Web Services and Applications (GWS) is a set of REST APIs and user interfaces that provide a web-based client interface to access Genesys services. Workspace Web Edition is the required desktop for this solution. The Genesys softphone or a Genesys supported SIP hardphone can be used for voice interactions with the agent. The agent can log into any data center as he or she is effectively logging into a single switch.
- Outbound Contact is an automated system that is used to create, modify, and run outbound dialing campaigns/dialing sessions in which agents interact with customers.
- Genesys Mobile Services (GMS) controls and exposes the Genesys API functionality to external applications by REST APIs, and provides critical callback services.
- Genesys Interaction Recording is a call recording solution, screen capture, and Quality Monitoring (QM) tool utilized to store, manage, and playback recorded voice conversations and screen captures, as well as provide quality assurance.

See Client connections in SIP Cluster and the diagram that illustrates component connectivity in the SIP Cluster.

What is a SIP Cluster Node

A SIP Cluster node instance is a set of components running on a computer host. An HA pair of SIP Cluster instances forms a SIP Cluster node.

Reference configuration of the SIP Cluster node contains the following components:

- SIP Server
- URS
- ORS
- Stat Server
- Interaction Concentrator

All components are deployed with corresponding redundancy. To increase call process capacity, deploy several SIP Cluster nodes.

Minimum Recommended Versions

The following table lists the components and their minimum recommended versions that are part of the SIP Cluster solution.

Component Name	Minimum Recommended Version
Genesys Info Mart	8.5.011.04
Genesys Interaction Recording	8.5.222.00
Genesys CX Insights (replaces GI2)	9.0.007.03
Genesys Mobile Services	8.5.112.05
GVP Media Control Platform	8.5.185.34
GVP Resource Manager	8.5.185.37
Interaction Concentrator	8.1.514.10
Management Framework	8.5.1+
Orchestration Server	8.1.400.67
Outbound Contact Server	8.1.509.06
SIP Feature Server	8.1.202.06
SIP Proxy	8.1.100.72
SIP Server	8.1.103.11

Stat Server	8.5.108.19
Universal Routing Server	8.1.400.45
Web Services and Applications	8.5.202.40

SIP Server in Cluster Mode

With SIP Server in Cluster mode, you can create a highly flexible architecture designed to utilize resources evenly by distributing calls and agent sessions across all available SIP Servers. Once configured, new SIP Servers added to the cluster start accepting calls and agent sessions automatically.

SIP Server in Cluster mode is designed to support high call volumes over a large number of SIP phones and logged-in agents.

When working in Cluster mode, SIP Server uses the following internal modules to provide SIP Cluster functionality:

- **Session Controller**: call processing engine that manages SIP signalling and T-Library protocol for calls, processed locally by SIP Server. Universal Routing Server (URS) and Orchestration Server (ORS) are the only clients that connect to the default port of Session Controller.
- **Interaction Proxy**: T-Library interface that distributes events for local calls grouped by a call interaction. Interaction Concentrator server (ICON) is the only client supporting the Interaction Proxy (IProxy) protocol.
- **T-Controller**: T-Library interface that maintains a subset of agent/DN states and interconnection to other T-Controllers in the cluster. Supports the standard T-Library protocol and the *smart client* T-Library protocol extension.
- **Smart Proxy**: T-Library interface for event monitoring clients (Stat Server). This module proxies communication between clients and all T-Controllers to increase SIP Cluster flexibility.

Client connections in SIP Cluster

T-Library clients can be grouped based on their connections in SIP Cluster, as follows:

- **Smart clients**—ICON, SIP Feature Server, custom smart clients—support the *smart client* T-Library protocol extension. Smart clients connect to the T-Controller port (TCport) of a Cluster node and receive events on DNs that are owned by this node.
- **Agent desktop clients**—Web Services and Applications (GWS), GPlus Adapters—support the standard T-Library protocol and send agent-related requests. Agent desktop clients connect to the T-Controller port (TCport) and receive events on registered DNs.
- **Legacy DN monitoring clients**—Stat Server—don't support the T-Controller's *smart client* protocol extension and don't send agent-related requests. DN monitoring clients connect to the Smart Proxy port (SmartProxy).
- **Local node routing clients**—URS, ORS—monitor and route calls processed by a local SIP Cluster node. Local node routing clients connect to the default port.

The following diagram presents component connectivity of a SIP Cluster Node with the Smart Proxy module enabled.

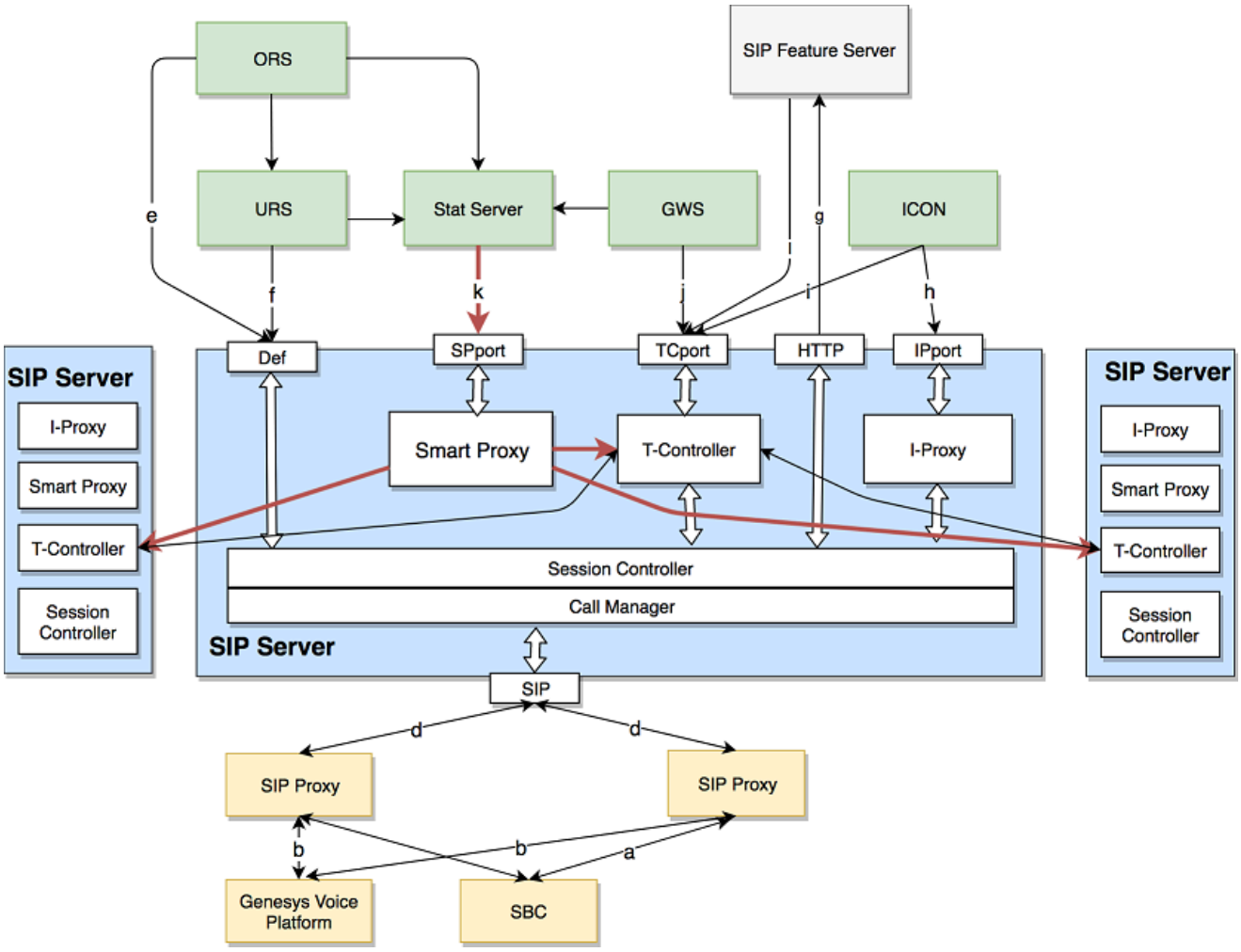


Diagram legend:

a. SIP: SBC load balances the incoming SIP traffic to SIP Proxies.
b. SIP: SIP Server communicates with Genesys Voice Platform (GVP) through SIP Proxy.
c. RTP: Media connection between the SBC and GVP.
d. SIP: SIP Server uses SIP Proxies in active-active mode.
e. TLib: ORS is connected to the default port of SIP Server to monitor Routing Points.
f. TLib: URS is connected to the default port of SIP Server to monitor Routing Points.
g. HTTP: SIP Server sends queries to an external Dial Plan implemented in SIP Feature Server.
h. TLib: ICON connects to the Interaction Proxy port (IPport) to receive call-related T-Library and call monitoring events.
i. TLib: ICON connects to the T-Controller port (TCport) to receive agent-related events.
j. TLib: Web Services and Applications (GWS) connects to the TCport to register to agent DNs as requested by Workspace Web Edition (WWE) desktops.

k. TLib: Stat Server connects to the SmartProxy port to register to all Extension and Routing Points DNs.

l. TLib: SIP Feature Server connects to the T-Controller port (TCport).

SIP Server Internal Modules

Session Controller

Session Controller (SC) is responsible for processing the call. SC operation is comparable to how SIP Server works in standalone mode. SC sends call-related events to its clients—URS and ORS—for the DNs that are involved in locally processed calls, on the node where those calls were received. URS and ORS are connected directly to the default SC port. URS and ORS are registered on the Routing Point DNs and receive information only about the calls processed on this SIP Server. This approach limits the load on URS and ORS and, as a result, improves the routing solution felxibility .

To process a higher call volume, you can change up the SIP Cluster by adding new instances of SIP Server. The SIP Cluster is designed to distribute all calls across all existing SC's. Each call is processed by one SC. All manipulations required for this call, such as transfers and conferences, are performed on this SC. A call is never transferred from one SC in the SIP Cluster to another. The SIP Cluster architecture ensures that the same SC processes all related calls, such as main and consultation calls initiated from the same DN.

Interaction Proxy

Interaction Proxy (IProxy) balances call-related events across multiple instances of ICON. In the high performance environment, the cluster of ICONs can be connected to one SIP Cluster node and calls processed on this node will be evenly distributed across all instances of ICON. All call-monitoring events and call-related T-Events generated for one call are directed to the same instance of ICON.

IProxy uses the IPport listening port to communicate with its clients.

T-Controller

T-Controller (TC) is responsible for the following actions:

- Maintaining the states of specific DNs and agents; generating events for these DNs and agents to the clients connected to the TCport and also to the other TCs in the SIP Cluster.
- Proxying T-Events received from other TCs in the SIP Cluster to the local clients connected to the TCport.
- Proxying call-related T-Events from a local SC to clients connected to the TCport and also to the other TCs in the SIP Cluster.

There are two types of T-Controller clients:

- TC provides a *T-Library-based smart client interface*. This interface allows TC clients to monitor only those DNs, that have an ownership on this TC. This approach optimizes lengthy and CPU/network-consuming registration process, which benefits both the server and its client. SIP Feature Server and

ICON are the examples of smart clients.

- TC also supports the *regular T-Library protocol* for backward compatibility to allow legacy clients to connect. Legacy clients connected to TC and registering all SIP Cluster DNs affect the SIP Cluster capacity, which degrades with the increasing number of legacy T-Library clients. For Stat Server applications, Genesys recommends enabling the Smart Proxy module.

T-Controller DN Ownership

All Extension DNs (and associated agents) are distributed across available SIP Cluster nodes for state maintenance. If SIP Cluster controls the state of a certain Extension DN, it means that SIP Cluster owns all activities associated with the DN, including its calls, a logged-in agent, and supervision subscriptions.

There are two types of DN ownership:

- *T-Library-based ownership* (DN contact is *not* set to "*") is for the PSTN/remote agents. For these DNs, ownership is assigned to the first SIP Server in the data center that receives a TRegisterAddress request from a client privileged to establish DN ownership. Ownership is released when the TUnregisterAddress request is received from the client for the DN or the client disconnects. The names of client applications that are privileged to establish DN ownership, are defined by the DN-level **dn-owner-applications** option , which is configured on the VoIP Service DN with **service-type**=`sip-cluster-nodes`.
- *SIP-based ownership* (the DN contact is set to "*") is for SIP registered phones. For these DNs, ownership is assigned to the first SIP Server in the data center that receives a SIP REGISTER request from SIP Proxy. Ownership is released after SIP REGISTER expires.

Smart Proxy

Introduced in SIP Server version 8.1.103.24, Smart Proxy offers the T-Library-based interface with a dedicated listening port to legacy T-Library clients, such as Stat Server, to monitor a large number of DNs regardless of the DN ownership by a Cluster Node. This greatly improves SIP Cluster flexibility. .

Smart Proxy connects as a smart (bulk-registrar) client to all SIP Cluster T-Controllers, enabling T-Controller optimization and reducing inter T-Controller message exchange. By opening a single connection to the Smart Proxy, T-Library monitoring clients receive information about all DNs.

Smart Proxy does not accept agent and supervisor requests, and smart T-Library connections. The agent desktop and bulk registrants must be connected to the T-Controller port.

The Smart Proxy module is enabled on the SIP Cluster Node by configuring the SmartProxy port in the SIP Server application and connecting Stat Server applications to that SmartProxy port instead of the T-Controller port.

> **Important**
>
> Genesys recommends enabling Smart Proxy for all new SIP Cluster deployments. For existing SIP Cluster deployments, Smart Proxy can be enabled if the T-Controller peak CPU usage reaches 75%.

Enabling Smart Proxy

To enable Smart Proxy functionality in the existing SIP Cluster environment, perform the following steps in all SIP Cluster Nodes. Each step must be completed in all SIP Cluster Nodes in all data centers, before continuing to the next step. Consider performing these steps during a maintenance window.

1. Upgrade all SIP Server applications to version 8.1.103.24 or later.
2. Configure the SmartProxy port in all SIP Server applications.
3. Restart SIP Server applications to enable Smart Proxy by using the rolling restart procedure, without service interruption:
    1. Restart the backup application.
    2. Wait for primary/backup synchronization to be completed.
    3. Do a switchover.
    4. Restart the new backup application.
4. Reconfigure Stat Server applications to connect to the SmartProxy port instead of the T-Controller port (TCport).
5. Restart Stat Server applications.
6. Set the **smart-proxy-enabled** option to `true` in the SIP Cluster Node DN. This triggers T-Controllers to work in optimized mode.

Disabling Smart Proxy

To disable Smart Proxy functionality:

1. Set the **smart-proxy-enabled** option to `false` in the SIP Cluster Node DN. This triggers T-Controllers to work in non-optimized mode.
2. Reconfigure Stat Server applications to connect to the T-Controller port (TCport) instead of the SmartProxy port.
3. Restart Stat Server applications.
4. Remove the SmartProxy port from SIP Server applications.
5. Restart SIP Server applications.

SIP Server logs in SIP Cluster mode

SIP Server in Cluster mode runs in multithreaded mode. The following prefixes represent the different subsystems in the SIP Server:

Thread ID	Example log name	Log tag	Subsystem	Thread name	Thread purpose
-	SIPS_usw1c_1.201.20160913_173417_999.log		Session Controller	Main thread	T-Library messages for the calls processed on a local node. It

Thread ID	Example log name	Log tag	Subsystem	Thread name	Thread purpose
					does not contain T-Library events related to an agent state.
001	SIPS_usw1c_1-001.20160913_173413_638.log	001	Session Controller	Call Manager	SIP processing. The log contains all call-related SIP messages.
512	SIPS_usw1c_1-512.20160913_173413_575.log	512	Session Controller	Service Checker	Manages OPTIONS messages and associated in-service/out-of-service messages.
768	SIPS_usw1c_1-768.20160913_173413_393.log	768	Session Controller	Transport	Connection for all SIP traffic, but the log does not contain much data.
1024	SIPS_usw1c_1-1024.20160913_173413_557.log	1024	Interaction Proxy	Interaction Proxy	Interaction Proxy subsystem. Contains call-based T-Library messages for ICON.
1280	SIPS_usw1c_1-1280.20160913_173413_703.log	1280	T-Controller	TController	T-Controller subsystem. Contains call and agent-related T-Library messages for all clients.
1536	SIPS_usw1c_1-1536.20160913_173413_822.log	1536	Session Controller	System Monitor	Shows collected statistics available through the HTTP interface.
1792	SIPS_usw1c_1-1792.20160923_172451_126.log	1792	Smart Proxy	Smart Proxy	Smart Proxy subsystem. Contains all T-Library messages for Stat Server.

See Configuring SIP Servers.

SIP Proxy

SIP Proxy provides an interface for SIP communication between SIP devices and SIP Server components. It handles register requests, load-balances SIP transactions between SIP Cluster nodes, and provides an alternative HA model that supports deploying primary/backup SIP Server instances as the HA pair across different subnets and does not require a virtual IP address. At least two SIP Proxies must be deployed for each data center. SIP endpoints (agent softphones, etc.) can either register directly with the SIP Proxy or an SBC, which then forwards the registration to the SIP Proxies.

SIP Registration

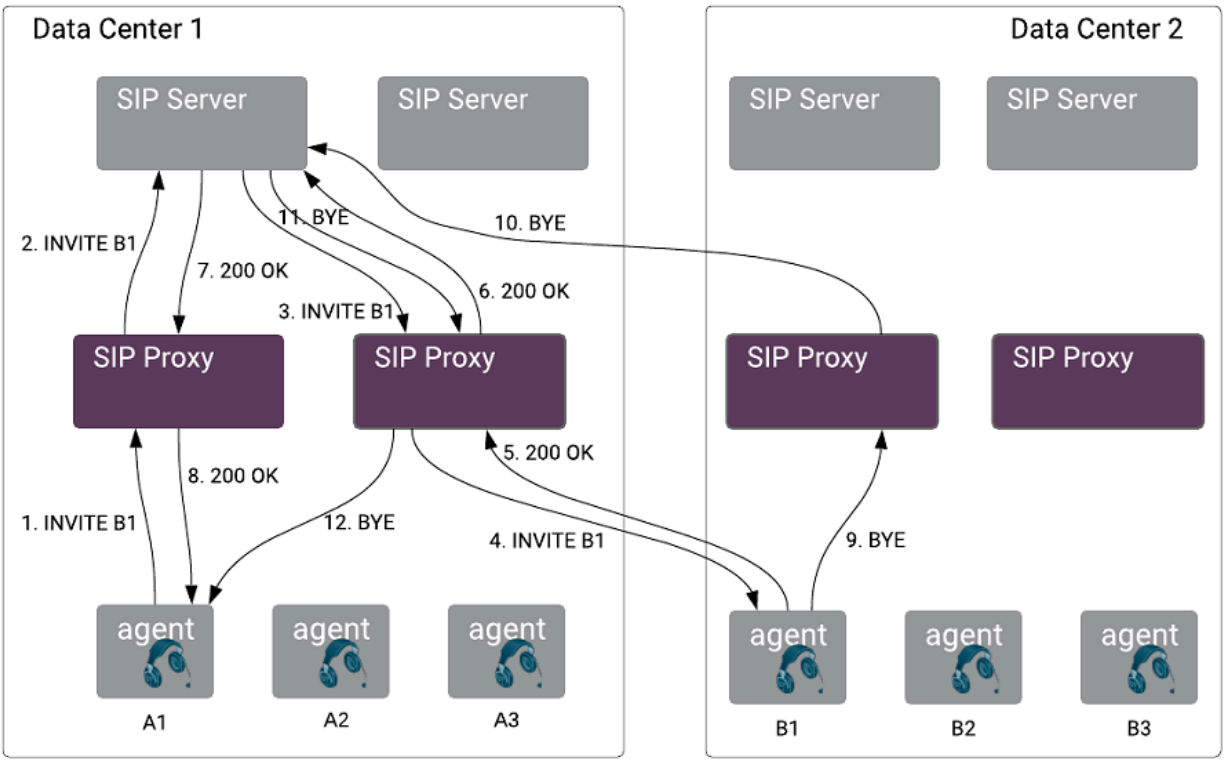
When operating in a cluster environment with one or more data centers, the set of SIP Proxy instances can be divided among the data centers. SIP Proxy obtains the lists of SIP Proxies and SIP Servers configured in the cluster from the SIP Cluster DN, that is configured in the Cluster Switch object. SIP Proxy uses the **geo-location** option to identify local SIP Servers that belong to a particular data center. Those SIP Servers contain the **geo-location** option with the same value as in the SIP Proxy configuration.

SIP Proxy acts as a SIP registrar. It has a shared registration-info storage. Any endpoint can be reached by any SIP Proxy. SIP Proxy uses SIP Server as an authentication server. It passes REGISTER requests to SIP Server and waits for a response. If it receives the 200 OK response, the registration is stored. All responses are always forwarded to the initiator.

SIP Proxy forwards all SIP messages from the endpoints to local SIP Servers. SIP Proxy can also forward outgoing messages to other data centers. Replication of registration and call data is performed across all SIP Proxy instances in the cluster.

SIP Proxy uses the Active out-of-service detection (**oos-check**) method for updating a list of active SIP Servers in a cluster and for detecting the primary SIP Server in an HA pair. SIP Proxy forwards SIP messages only to the primary SIP Server of the HA pair.

Call flow example with two data centers:



See Configuring SIP Proxy.

Unsupported and Partially Supported Functionality

> **Important**
> SIP Cluster solution is under restricted availability. Contact Product Management for more information.

This topic provides high-level information about functionality that is not supported or partially supported in this release of SIP Cluster. Consult documentation of particular products and corresponding product teams for possible limitations of their products in the SIP Cluster architecture.

Solution	Limitations or differences
SIP Server	Certain features or functionality are either not supported in SIP Cluster or are not fully supported. For full details, see: • Unsupported SIP Server functionality • Unsupported SIP Server configuration options • Partially supported SIP Server functionality
Universal Routing	The following Universal Routing functionality is not supported in SIP Cluster: • Certain functions and objects, see Routing Limitations • Some predefined statistics and predefined macros • Strategy for ring-no-answer situations • Cost-based routing • Load balancing • Agents participating in multiple outbound campaigns
Historical Reporting	All Interaction Concentrator and Genesys Info Mart functionality is supported in SIP Cluster, but certain data is no longer reported or is reported differently. For more information, see Historical Reporting Deployment Considerations.
Web Services and Applications	Web Services and Applications (GWS):

Solution	Limitations or differences
	• Complete SIP Cluster integration and support will be implemented and documented in GWS version 9.0. The existing version of GWS 8.5 is also available for SIP Cluster deployment and requires a Product Management approval. The following Workspace Web Edition 8.5.2 on-premises features are not supported for SIP Cluster: • Remote agent phone number support • Nailed up connection establishment on first call • Routing Point monitoring and coaching
Genesys Interaction Recording (GIR)	The GIR solution supports SIP Cluster premise deployments, but with limitations where features cannot be configured at the Agent Login object level, such as wrap-up-time and Full-time Recording.

Unsupported SIP Server functionality

The following SIP Server functionality is not supported in SIP Cluster:

- ACD Queues
- ISCC (Multi-site support)
- Alternate Routing - Stranded Calls
- Associating an ACD Queue with a Routing Point
- Asterisk Voice Mail Integration
- Call Completion Features
- Call Park/Retrieve
- Call Pickup
- Call Recording NETANN-Based
- Supervision of Routing Points (IVR supervision supported instead)
- Remote Supervision
- Class of Service
- Dummy SDP
- E911 Emergency Gateway
- Find Me Follow Me
- Hunt Goups
- IMS Integration
- Instant Messaging
- Media Server Reliability NETANN
- Nailed-Up Connections
- P-Access-Network-Info Private Header
- Presence from Switches and Endpoints
- Preview Interactions
- Remote Server Registration
- Shared Call Appearance
- Smart OtherDN Handling
- Trunk Capacity Support

Unsupported SIP Server configuration options

This list contains SIP Server configuration options that are not supported and not related to the functionality listed above.

Application level:

- dtmf-payload
- external-registrar
- emergency-recording-cleanup-enabled
- emergency-recording-filename
- internal-registrar-enabled
- internal-registrar-domains
- internal-registrar-persistent
- max-legs-per-sm
- registrar-default-timeout
- shutdown-sip-reject-code
- sip-legacy-invite-retr-interval
- sip-retry-timeout

Partially supported SIP Server functionality

The following table presents SIP Server functionality partially supported in SIP Cluster. The table entries use these notations:

- N—Not supported
- Y—Supported
- P—Partially supported
- App—Application-level setting
- DN—DN-level setting
- AL—Agent Login-level setting
- Ext—AttributeExtensions
- RP—Routing Point

Feature Name	Setting Level	Supported	Comments
Alternate Routing			
Alternate Routing - Unresponsive ORS/URS		P	Alternate routing via ISCC is not supported. A call can be redirected only within a SIP Cluster node receiving a call.
sip-invite-timeout	App	Y	

Feature Name	Setting Level	Supported	Comments
no-response-dn	DN	P	It works but conflicts with SIP Feature Server forwarding.
default-dn	App, DN (RP)	Y	
router-timeout	App	Y	
default-route-point	App	Y	
Automatic Inactive Agent Logout			
auto-logout-timeout	App, DN (RP)	P	Only on the Application level is supported.
auto-logout-ready	App, DN (RP)	P	Only on the Application level is supported.
logout-on-disconnect	App, DN (RP)	P	Only on the Application level is supported.
Call Recording MSML-based			
recording-filename	App	Y	
msml-record-support	App	Y	
record-consult-calls	App	Y	
record	DN	P	Only on the DN level is supported, but on the Agent Login level is not.
Dynamic Call Recording			
id	Ext	N	
record	Ext	P	It is supported only in TRouteCall.
dest	Ext	N	
params	Ext	N	
Emulated Agents			
emulate-login	Ext	N	
emulated-login-state	App, DN	P	Only on the Application level is supported.
agent-strict-id	App	N	
sync-emu-agent	App, DN	N	
override-switch-acw	App	N	
untimed-wrap-up-value	App	N	
wrap-up-time	App, DN, Ext	P	Only on the Application level and in AttributeExtensions are supported.
wrap-up-threshold	App	N	
legal-guard-time	App, Ext	Y	
timed-acw-in-idle	App	Y	

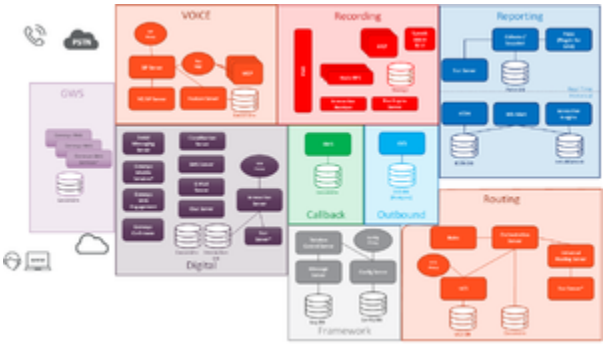
Feature Name	Setting Level	Supported	Comments
acw-in-idle-force-ready	App	Y	
agent-emu-login-on-call	App, DN, Ext	N	
agent-logout-on-unreg	App, Ext	N	
enable-agentlogin-presence	App	N	Implemented in SIP Feature Server instead.
enable-agentlogin-subscribe	App	N	Implemented in SIP Feature Server instead.
auto-logout-ready	App, DN	P	Only on the Application level is supported.
auto-logout-timeout	App, DN	P	Only on the Application level is supported.
logout-on-disconnect	App, Ext	Y	
logout-on-out-of-service	App	Y	
reason-in-extension	App	P	It is triggered by NOTIFY. But SIP Cluster does not know the agent state.
agent-logout-reasssoc	App	N	TRegisterAddress from the same client name does not re-associate ownership, only TAgentLogin does.
Endpoint Service Monitoring			
oos-check	DN	Y	
oos-force	DN	Y	
recovery-timeout	DN	Y	
oos-options-max-forwards	DN	Y	
sip-oos-enabled	DN	N	Passive OOS is not supported.
No-Answer Supervision			
*-no-answer-overflow	App, DN, AL	P	agent-no-answer-overflow -- Only on the Application level is supported.
*-no-answer-action	App, DN, AL	P	agent-no-answer-action -- Only on the Application level is supported.
*-no-answer-timeout	App, DN, AL	P	agent-no-answer-timeout -- Only on the Application level is supported.
nas-private	App	Y	
set-notready-on-busy	App	Y	
NO_ANSWER_TIMEOUT	Ext	Y	
NO_ANSWER_OVERFLOW	Ext	Y	
NO_ANSWER_ACTION	Ext	Y	

- Multi-Site Supervision--Supervision works when a supervisor is in SIP Cluster and an agent is in the standalone switch. It does not work when a supervisor is on the standalone switch and an agent is in SIP Cluster.

Deploying SIP Cluster

> **Important**
> SIP Cluster solution is under restricted availability. Contact Product Management for more information.

The SIP Cluster for Premise deployments typically involve two or more data centers in geographically distinct locations. Some components are designed to operate in a geo-dispersed manner while others need to be deployed in more traditional local or disaster recover modes. The Cluster deployment involves several different areas or layers of functionality. The following diagram depicts those deployment layers.



Cluster Deployment Layers (click to expand)

Task Summary

The following task summary provide an overview of the steps that are required to deploy the SIP Cluster solution:

1. Complete prerequisites, including planning, hardware, and installation of Management Framework software and supported applications.
2. Plan and configure DNS records.
3. Verify that you have the required minimum Genesys component versions.
4. Configure Genesys Voice Platform components.
5. Configure Switch and DN objects.
6. Configure the cluster core components:
   - SIP Server
   - SIP Proxy

- SIP Feature Server
- Stat Server
- URS and ORS

7. Configure Virtual Queue (VQ) components:
   - Switch and DN objects
   - SIP Server
8. Deploy the supported applications that you selected during the planning stage, which might include:
   - Historical Reporting
   - Outbound Contact
   - Web Services and Applications
   - Genesys Mobile Services

Prerequisites

> **Important**
> SIP Cluster solution is under restricted availability. Contact Product Management for more information.

Before configuring your SIP Cluster environment, you must complete the following tasks:

1. Plan and set up the required hardware and network to serve the size, features, and distribution of your data and call centers. Review requirements below.
2. Deploy Genesys Management Framework. Genesys recommends deploying Framework for Disaster Recovery/Business Continuity scenarios. Ensure that application components use Configuration Server Proxy in the given data center, instead of the master Configuration Server, to obtain their configuration.
3. Deploy Genesys Administrator Extension.
4. Deploy and configure all required supported applications and any optional supported applications you have selected.

OS Requirements

Linux and Windows are supported operating systems for the SIP Cluster environment.

Java Requirements

See topics of the respective components that use Java JDK for detailed information on its configuration.

Cassandra Requirements

SIP Cluster requires that your environment includes Cassandra 2 or later. Genesys recommends Cassandra version 2.2. See topics of the respective components that use Cassandra for detailed information on its configuration.

Configuring DNS Records

> **Important**
> SIP Cluster solution is under restricted availability. Contact Product Management for more information.

SIP Cluster SIP connectivity relies on SIP Cluster Fully Qualified Domain Name (FQDN) with SRV-based name resolution.

SRV-type records for the SIP Cluster are registered in a geo-aware DNS zone. The same SIP Cluster FQDN is used in each data center. The same priority, weight, and port must be specified within the SRV records.

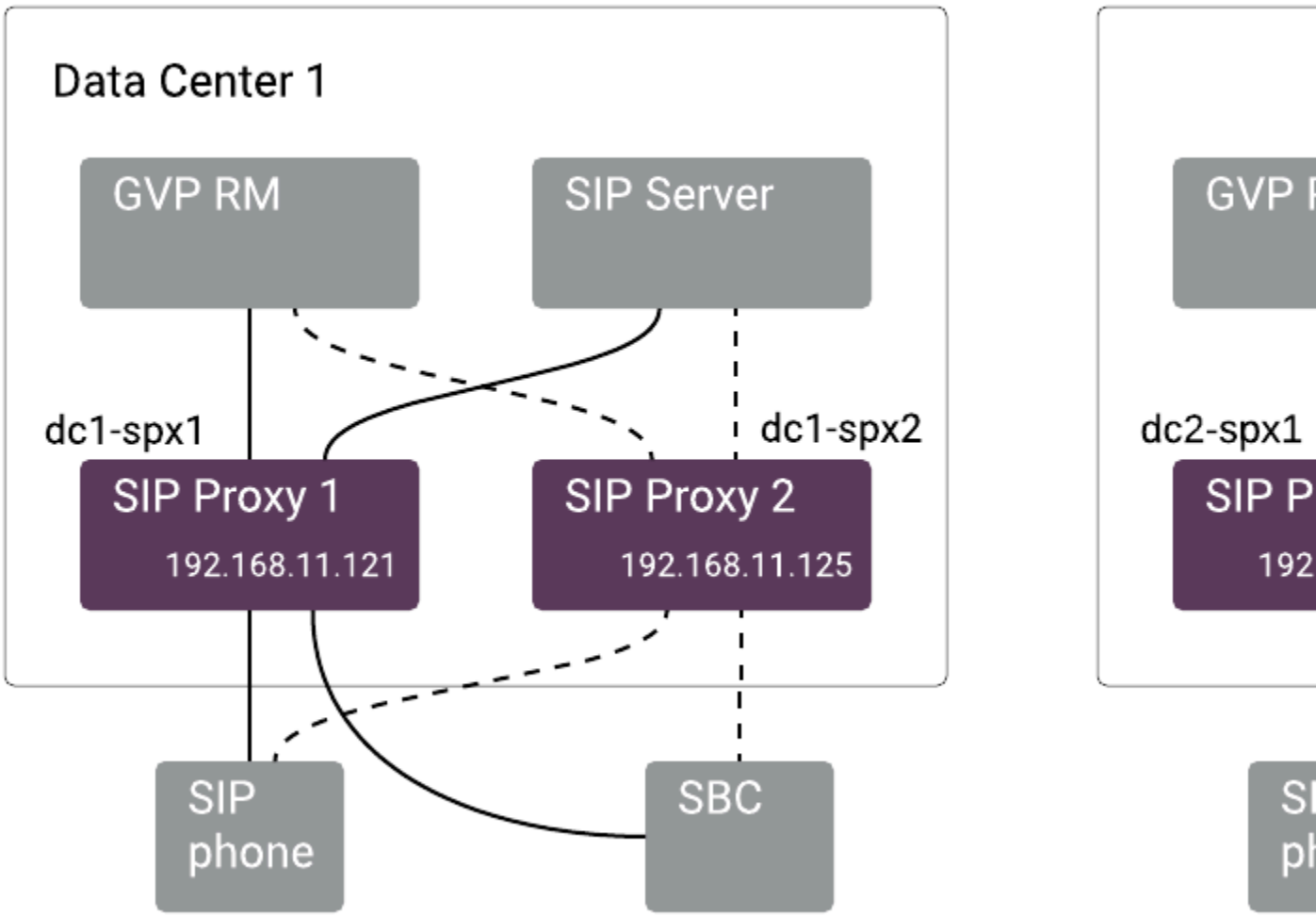
All SIP Proxies in the SIP Cluster use the same port number (**sip-port**) for SIP signaling. Each SIP Proxy runs on its own machine and uses its own IP address. For a given data center, the SRV records resolve into the IP addresses of the SIP Proxies located in this data center and the port number used by all SIP Proxies. SIP Server, Resource Managers, the Session Border Controller (SBC), and SIP Phones distribute SIP requests across the SIP Proxies according to the DNS resolution and in the load-balancing manner that they support. See the SIP Proxy IP address and DNS mapping example below.

A SIP Phone or SBC can distribute requests across all local operational proxies, or can persist with one SIP Proxy as long as the SIP Proxy is operational.

Required Provisioning

- The Host name on which SIP Proxy is installed: The FQDN specified as the target within the Record Data of the SRV records should be identical to the FQDN specified as the Host name of the Host object. If the Host object uses an IP address and does not use an FQDN, the FQDN specified within the SRV record must resolve into this IP address specified as the Host name of the Host object.
- The **sip-port** of the SIP Proxy application: The port number specified within the Record Data of the SRV records must be identical to the port number specified in the **sip-port** parameter on the Server Info tab of the SIP Proxy application.
- The FQDN that resolves to the SRV list of the SIP Proxy addresses is specified in the SIP Outbound Proxy DN configuration, including transport:
  - **contact**
  - **external-contact**
- If the SBC or SIP Phones do not support SRV name resolution, specify another FQDN in the **external-contact** option. This FQDN should be registered as an Address-type record and should be resolved using A or AAAA type queries.

Example: SIP Proxy IP address and DNS mapping



DNS record mapping example (click to expand)

In data center 1 (DC1), SIP Cluster SIP connectivity as follows:

- SIP Proxies:
  - DC1 SIP Proxy 1: 192.168.11.121
  - DC1 SIP Proxy 2: 192.168.11.125
- DNS records for the SIP Proxy machines:

- **dc1-spx1**.example.com. IN A 192.168.11.121
- **dc1-spx2**.example.com. IN A 192.168.11.125

In data center 2 (DC2), SIP Cluster SIP connectivity as follows:

- SIP Proxies:
  - DC2 SIP Proxy 1: 192.168.22.121
  - DC2 SIP Proxy 2: 192.168.22.125
- DNS records for the SIP Proxy machines:
  - **dc2-spx1**.example.com. IN A 192.168.22.121
  - **dc2-spx2**.example.com. IN A 192.168.22.125

As shown in the diagram, the SBC and GVP Resource Manager send new SIP requests to the local SIP Cluster SIP Proxies. Geo-aware name resolution enables local SIP signaling of the SIP requests delivered to the SIP Cluster.

SRV records for SIP Cluster: UDP protocol example

- SIP Proxies listen on port **5060**
- The SIP Cluster FQDN: **spx-srv**.example.com

For the systems that prefer SIP signaling through the SIP Proxy located in data center 1 and use the UDP protocol for SIP signaling (usually phones and the SBC), the geo-aware name zone should include SRV records as follows:

- _sip._udp.**spx-srv**.example.com. IN SRV 10 50 5060 **dc1-spx1**.example.com.
- _sip._udp.**spx-srv**.example.com. IN SRV 10 50 5060 **dc1-spx2**.example.com.
- _sip._udp.**spx-srv**.example.com. IN SRV 200 50 5060 **dc2-spx1**.example.com.
- _sip._udp.**spx-srv**.example.com. IN SRV 200 50 5060 **dc2-spx2**.example.com.

For the systems that prefer SIP signaling through the SIP Proxy located in data center 2 and use the UDP protocol for SIP signaling (usually phones and the SBC), the geo-aware name zone should include SRV records as follows:

- _sip._udp.**spx-srv**.example.com. IN SRV 10 50 5060 **dc2-spx1**.example.com.
- _sip._udp.**spx-srv**.example.com. IN SRV 10 50 5060 **dc2-spx2**.example.com.
- _sip._udp.**spx-srv**.example.com. IN SRV 200 50 5060 **dc1-spx1**.example.com.
- _sip._udp.**spx-srv**.example.com. IN SRV 200 50 5060 **dc1-spx2**.example.com.

SRV records for SIP Cluster: TCP protocol example

- SIP Proxies listen on port **5060**
- The SIP Cluster FQDN: **spx-srv**.example.com

For the systems that prefer SIP signaling through the SIP Proxy located in data center 1 and use the TCP protocol for SIP signaling (usually SIP Server and GVP RM), the geo-aware name zone should include SRV records as follows:

- _sip._tcp.**spx-srv**.example.com. IN SRV 10 50 5060 **dc1-spx1**.example.com.
- _sip._tcp.**spx-srv**.example.com. IN SRV 10 50 5060 **dc1-spx2**.example.com.
- _sip._tcp.**spx-srv**.example.com. IN SRV 200 50 5060 **dc2-spx1**.example.com.
- _sip._tcp.**spx-srv**.example.com. IN SRV 200 50 5060 **dc2-spx2**.example.com.

For the systems that prefer SIP signaling through the SIP Proxy located in data center 2 and use the TCP protocol for SIP signaling (usually SIP Server and GVP RM), the geo-aware name zone should include SRV records as follows:

- _sip._tcp.**spx-srv**.example.com. IN SRV 10 50 5060 **dc2-spx1**.example.com.
- _sip._tcp.**spx-srv**.example.com. IN SRV 10 50 5060 **dc2-spx2**.example.com.
- _sip._tcp.**spx-srv**.example.com. IN SRV 200 50 5060 **dc1-spx1**.example.com.
- _sip._tcp.**spx-srv**.example.com. IN SRV 200 50 5060 **dc1-spx2**.example.com.

SIP registration in failover scenarios based on SRV records

Create DNS SRV records for your SIP Cluster as described in the previous section. The components that communicate with the SIP Proxy directly use this FQDN.

If your SIP phones communicate with the SIP Cluster via SBCs, define an additional FQDN of the SIP Cluster and an FQDN for each SBC. The IP address of each SBC is defined by your telecom team. For each SBC, create A-type record that resolves the SBC FQDN into the IP address of this SBC.

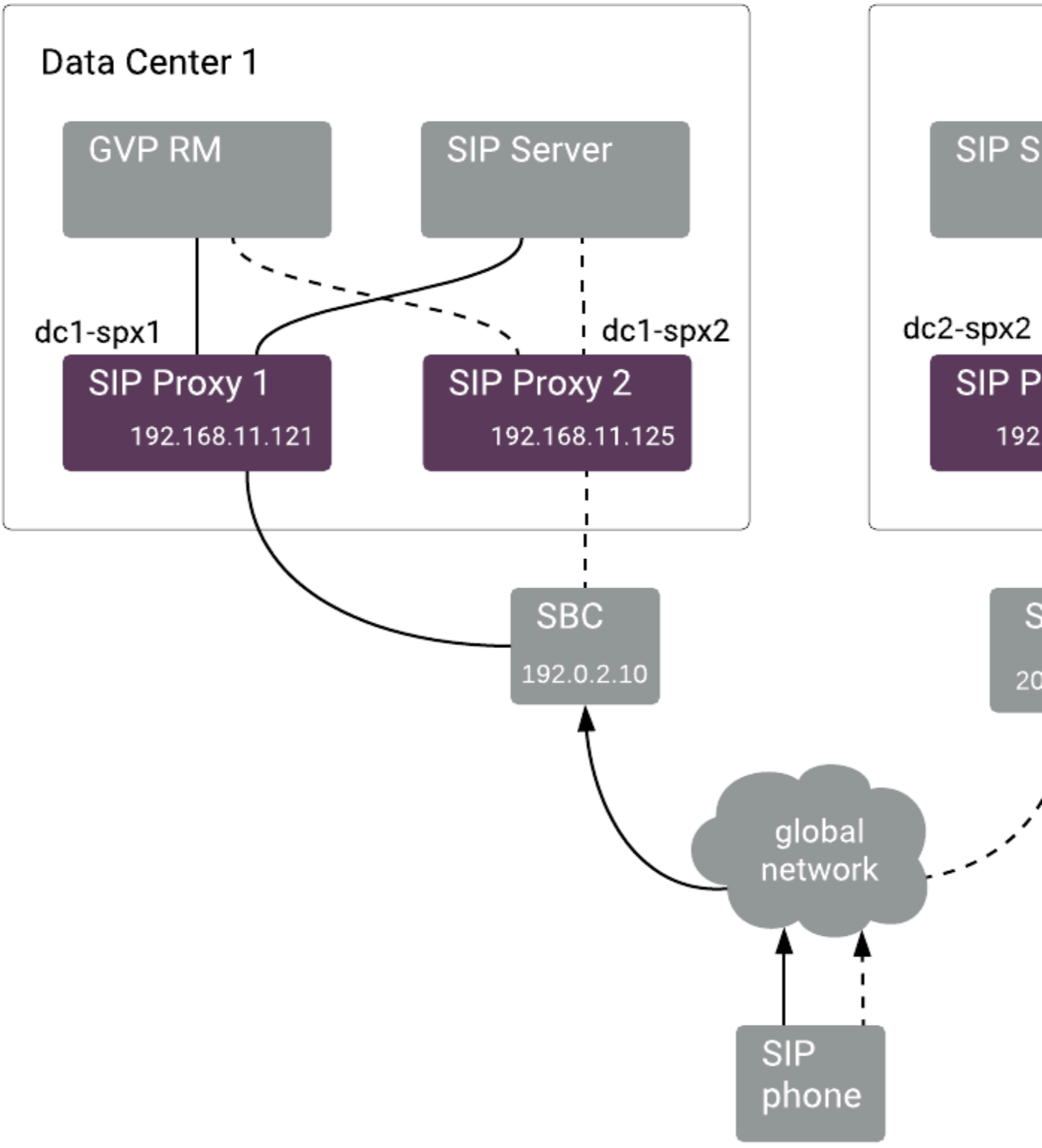
Create SRV records that the phones will use, as follows:

- Specify FQDNs of the SBC in the Record Data of the SRV records.
- Configure priority and weight within the SRV records according to data center priorities in your environment.

The SRV records priorities instruct the phones to establish SIP registration with the primary data center and use the second data center as last resort. Each SIP phone attempts to register with the primary data center. If the primary data center is inaccessible, the phone registers with the secondary data center.

A phone keeps renewing its registration with a data center, either the primary or the secondary one. The phone will re-attempt registration with the primary data center when one of the following occurs:

- the phone is restarted
- the secondary data center is no longer accessible
- the new agent’s login session is initiated (WWE starts the phone on agent login when configured in a Connector mode)


Data Center 1
GVP RM
SIP Server
dc1-spx1
dc1-spx2
SIP Proxy 1
192.168.11.121
SIP Proxy 2
192.168.11.125
SBC
192.0.2.10
global network
SIP phone
dc2-spx2

Failover scenario (click to expand)

IP address and SRV records example

Primary data center 1 connectivity:

- SBC IP address: 192.0.2.10
- DNS record for the SBC: **sip-dc1**.example.com. IN A 192.0.2.10

Recovery data center 2 connectivity:

- SBC IP address: 203.0.113.10
- DNS record for the SBC: **sip-dc2**.example.com. IN A 203.0.113.10

FQDN for SRV-based name resolution: sip-dc.example.com

- Best priority path (10) is via data center 1 SBC (port 6000) would be:
  _sip._udp.sip-dc.example.com. IN SRV **10** 50 6000 **sip-dc1**.example.com.
- Lower priority path (20) is via data center 2 SBC (port 6000) would be:
  _sip._udp.sip-dc.example.com. IN SRV **20** 50 6000 **sip-dc2**.example.com.

Provisioning for non-geo-aware DNS

If a geo-aware DNS is not available in a SIP Cluster deployment, the SBC must be configured to forward SIP REGISTER requests to SIP Proxy instances in only *one* data center, co-located with the SBC.

Configuring Switch and DN Objects for SIP Cluster

> **Important**
> SIP Cluster solution is under restricted availability. Contact Product Management for more information.

Complete these steps to create configuration objects required for the SIP Cluster environment.

1. Configure the SIP Cluster Switch
2. Configure the SIP Cluster Node DN
3. Configure the SIP Outbound Proxy DN
4. Configure MSML DNs
5. Configure Softswitch DNs
6. Configure Trunk DNs
7. Configure Routing Point DNs
8. Configure Extension DNs
9. Configure the Switch and DNs for Virtual Queues

Keep in mind the following:

- Agent Logins should not be configured in the SIP Cluster environment.
- All SIP Server applications representing SIP Cluster nodes must use the same switch.
- ACD Queue DNs are not supported in the SIP Cluster.

Configuring the SIP Cluster Switch

Create a **Switch** object of type **SIP Switch** dedicated to the SIP Cluster. with the name, for example, **SIP_Cluster**.

Configuring the SIP Cluster Node DN

The SIP Cluster Switch must contain a SIP Cluster Node DN. All SIP Server and SIP Proxy applications in the Cluster use the parameters configured in this DN.

1. Under **Switches** > **SIP_Cluster** > **DNs**, create a DN of type **Voice over IP Service** named, for example, **SIP_Cluster_DN**.
2. In the **Annex** > **[TServer]** section, configure the following mandatory options:

Name	Value	Notes	Example
addp-trace	full	The trace level for ADDP messages.	
addp-remote-timeout	11	T-Controller ADDP protocol setting.	
addp-timeout	7	T-Controller ADDP protocol setting.	
agent-state-auto-restore	false	Enables restoration of agent states when the DN ownership changes between SIP Cluster nodes. When set to `false`, agents are logged out on the DN ownership change and must log in manually. When set to `true`, agent states are restored on the new primary SIP Cluster node and agents are logged in automatically. **Limitation:** When an agent manually sets to NotReady state with ACW mode, SIP Server generates the ManualSetACWPeriod reason code automatically and does not restore the agent state on the new agent state owner (the SIP Cluster node).	
applications	<A comma-separated list of *primary* SIP Server application names in the cluster>	All SIP Server instances, except dummy SIP Server applications used by ICON, for all data centers.	**applications**=SIPS_sfo_1,SI
dn-owner-applications	<A comma-separated list of application names that can take DN ownership for a node>	The application name is sent to SIP Server as AttributeApplicationName in TRegisterClient. This name is used to identify a desktop and maintain DN ownership for remote agents. Usually, it is set to the name used by Genesys Web Services to register with SIP Server.	**dn-owner-applications**=Cluster
replace-agent-phone	false	When set to `true`, enables modification of dial digits that are required to reach an agent	

Name	Value	Notes	Example
		DN. This feature supports remote agents to use external numbers that are not provisioned in the Configuration Database. This option setting affects all cluster internal DNs. **Feature limitations:** • Dial plan rules are applied to the original DN object. The agent-phone digits are not processed by the dial plan and are not modified. • A 1pcc call coming from an external DN while it is engaged in a Login Session with agent-phone functionality is considered to be inbound. There is no event message regarding this party and the DN state is unknown. • SIP Server does not use a non-provisioned phone number to engage a supervisor in a monitored call because the dial plan is not applied when the supervisor is engaged in a monitored call.	
service-type	sip-cluster-nodes		
sip-enable-strict-auth	false	When set to `true`, enables SIP Server in SIP Cluster mode to mandate authorization of internal devices on REGISTER and INVITE requests. To register or establish communication, devices must not use empty passwords or passwords equal to the DN name. When this option is set to `false`, an internal device can register or establish communication with SIP Cluster without any authorization. You can define the **sip-enable-strict-auth** option at the following levels listed in order of priority: 1. DN of type Extension that is not behind a softswitch	

Name	Value	Notes	Example
		2. SIP Server Application 3. This SIP Cluster Node VOIP Service DN	
sipp-oos-recheck	false	When set to `true`, enables the Active Out-of-Service (OOS) check procedure for a VOIP service or Trunk DN to ensure that this DN is placed out of service only when SIP Proxy processes the SIP traffic. If set to `true`, SIP Server places the DN out of service only after rechecking that SIP Proxy is available, and does not place any other DN out of service if the SIP Proxy DN is already out of service. If set to `false` (the default), SIP Server behavior is not changed. As soon as the SIP Proxy DN is placed out of service, SIP Server initiates a switchover if the **switchover-on-sipp-oos** option is set to `true`. This enables establishing a connection to SIP Proxy from another HA-peer application.	
sipproxy-applications	<A comma-separated list of all SIP Proxy application names in the SIP Cluster>	All SIP Proxy instances for all data centers.	**sipproxy-applications**=SIPProxy_sfo_
smart-proxy-enabled	false	Set to `true` only when the Smart Proxy module is enabled and all Stat Server applications are connected to the SmartProxy port. Setting to `true` optimizes event distribution between T-Controllers and improves SIP Cluster performance. If the option is set to `false`, T-Controllers work in normal mode and Smart Proxy modules, if enabled, work in compatibility mode.	
switchover-on-sipp-oos	false	If set to `true`, SIP Server attempts to initiate a switchover immediately after the SIP Proxy VOIP DN is placed out of service. The switchover is initiated by the SERVICE_UNAVAILABLE message sent to LCA/SCS. When SIP Server is switched to backup mode by SCS, SIP Server issues	

Name	Value	Notes	Example
		the SERVICE_AVAILABLE message to SCS. The **sipp-oos-recheck** option must be set to `true`.	
tc-reconnect-timeout	0-60	Specify the timeout, in seconds, during which T-Controller tries to reconnect to another SIP Cluster node when the connection between SIP Cluster nodes is lost. When the timeout expires and the connection is not restored, DNs owned by the disconnected T-Controller are declared out of service. The default value of 0 (zero) disables this functionality.	
tc-latency-poll-interval	10	See Troubleshooting T-Controllers communication issues for details.	

Configuring the SIP Outbound Proxy DN

The SIP Cluster Switch must contain one SIP Outbound Proxy DN. All SIP Server and SIP Proxy applications in the cluster use the parameters configured in this DN.

1. Under **Switches** > **SIP_Cluster** > **DNs**, create a DN of type Voice over IP Service named, for example, **SIPProxy_DN**.
2. In the **Annex** > **[TServer]** section, configure the following mandatory options:

Name	Value	Notes	Example
contact	<The SIP access point of SIP Cluster (geo-aware SRV FQDN resolved to data center's SIP Proxy instances) with reliable transport>	Set this option to the Fully Qualified Domain Name (FQDN) that resolves to the SRV list of the SIP Proxy addresses. SIP Server uses this list to load-balance the traffic across all SIP Proxy instances available in a data center that is defined by **geo-location**. If there are multiple data centers within the cluster, this FQDN must be resolved to SIP Proxy instances located in the same data center with SIP Server, which is resolving the SRV FQDN. In other words, SIP Server must be aware only of those SIP Proxy instances installed in the same data center where SIP Server resides. It is	**contact**=`spx-srv.example.com;transport=`

Name	Value	Notes	Example
		the responsibility of the DNS to make this differentiation. This contact must be configured with the TCP transport to allow passing call attached data in SIP messages to GVP.	
external-contact	<SIP access point of SIP Cluster (geo-aware SRV FQDN resolved to data center's SIP Proxy instances)>	SIP Proxy sends the value of external-contact to its clients (e.g. GVP RM, SBC, SIP phones). SIP Cluster clients should be able to support the SRV FQDN.	**external-contact**=spx-srv.example.com
oos-check	10	These two options enable active out-of-service detection. In this specific example, SIP Server is checking for SIP Proxies (by using the value of the **contact** option) every 10 seconds and sets them in the out-of-service state 2 seconds after the last check for which a response was not received.	
oos-force	2		
service-type	sip-outbound-proxy		

Configuring MSML DNs

1. Under **Switches** > **SIP_Cluster** > **DNs**, create a DN of type **Voice over IP Service** for each data center containing contacts of all Resource Managers in the environment.
2. In the **Annex** > **[TServer]** section, configure the following configuration options:

Name	Value	Notes	Example
service-type	msml		
contact	<RM-SRV-FQDN>;transport=tcp	The SRV FQDN resolved to a pair of A-record FQDNs or IP addresses pointing to two Resource Managers configured as an active-active HA pair which are deployed to serve a particular data center.	**contact**=sfosrv-rm.example.com;transport=t
geo-location	<String>	A string identifying the data center to which this DN belongs.	**geo-location**=sfo
cpd-capability	mediaserver		
make-call-rfc3725-flow	1		
oos-check	10		
oos-error-check	true		

Name	Value	Notes	Example
oos-force	2		
prefix	msml=		
refer-enabled	false		
ring-tone-on-make-call	false		
sip-uri-params	gvp-tenant-id=<IVR_Profile_for_DC>	SIP Server sends the name of the IVR profile to be used to Resource Manager as a parameter of the INVITE's Request-URI header. The IVR profile will be used both for media services and recording. One IVR Profile must be created for each data center.	**sip-uri-params**=gvp-tenant id=sfo-15
subscription-id	Environment		
userdata-map-filter	*		
userdata-map-format	sip-headers-encoded		

Configuring Softswitch DNs

Softswitch DNs are mandatory for remote agents.

1. Under **Switches** > **SIP_Cluster** > **DNs**, create a DN of type **Voice over IP Service** for each data center in the SIP Cluster environment.
2. In the **Annex** > **[TServer]** section, configure the following mandatory options:

Name	Value	Example
contact	<SBC-FQDN or IP-address>:<Agent-SIP-port> Note: The contact pointing to the SBC realm used to access PSTN agents.	**contact**=sbc-example.com:5060
dual-dialog-enabled	false	
geo-location	<A string identifying the data center to which this DN belongs>	**geo-location**=sfo
make-call-rfc3725-flow	1	
oos-check	10	
oos-force	2	
oos-options-max-forwads	true	

Name	Value	Example
record	true	
refer-enabled	false	
service-type	softswitch	
sip-error-conversion	408=486	
sip-proxy-headers-enabled	false	
sip-ring-tone-mode	1	

Configuring Trunk DNs

1. Under **Switches** > **SIP_Cluster** > **DNs**, create a DN of type **Trunk** for each data center configured for a particular tenant.
2. In the **Annex** > **[TServer]** section, configure the following configuration options:

Name	Value	Notes	Example
contact	<SBC-FQDN or IP-address>:<SIP-port>	The contact pointing to the SBC realm.	**contact**=`sbc-example.com:5060`
geo-location	<String>	A string identifying the data center to which this DN belongs.	**geo-location**=`sfo`
dual-dialog-enabled	false		
oos-check	10		
oos-error-check	true		
oos-force	2		
oos-options-max-forwards	5		
prefix		This option can be used the same way as in a standard SIP Server deployment.	+
priority		This option can be used the same way as in a standard SIP Server deployment.	1
refer-enabled	false		
sip-proxy-headers-enabled	false		
sip-ring-tone-mode	1		

Configuring Routing Point DNs

Under **Switches** > **SIP_Cluster** > **DNs**, configure a DN of type **Routing Point** to be used for default routing in the SIP Cluster. Configure other Routing Point DNs as required for your environment.

Configuring Extension DNs

1. Under **Switches** > **SIP_Cluster** > **DNs** >, create DNs of type **Extension** for all agent phones for a particular tenant.
2. For agents using PSTN phones, configure extension DNs by specifying their phone numbers.
3. For agents using SIP phones, in the **Annex** > **[TServer]** section, configure the following configuration options:

Name	Value
contact	*
dual-dialog-enabled	true
make-call-rfc3725-flow	1
refer-enabled	false
use-register-for-service-state	true
use-contact-as-dn	true
sip-ring-tone-mode	1
sip-cti-control	talk,hold (or talk,hold,dtmf)

Configuring a Switch and DNs for Virtual Queues

Large contact center deployments may generate significant traffic on the Virtual Queue (VQ) DNs to satisfy complex reporting and monitoring requirements. In the SIP Cluster architecture, this traffic is isolated on a dedicated VQ switch served by several HA pairs of SIP Servers running in a non-cluster mode with one pair deployed in each data center. SIP Server instances serving the VQ Switch are also used for agent reservation.

Create a **Switch** object of type **SIP Switch** dedicated to Virtual Queue DNs with the name, for example, **VQ-switch**. Under the VQ-switch, configure DNs of type **Virtual Queue**. Only DNs of this type must be configured under this VQ-switch.

You will assign this VQ Switch to a SIP Server pair dedicated to VQ DNs.

Configuring SIP Servers

You must configure SIP Server applications for the following purposes:

- Cluster nodes
- Virtual queues
- Historical reporting

Configuring SIP Servers for SIP Cluster

1. Deploy SIP Servers as an HA pair, Hot Standby redundancy mode, by following the standard procedure.
   - Suggested application names: **SIPS_<datacenter>_1**, **SIPS_<datacenter>_1_B**.
2. On the **Switches** tab, add the SIP Cluster Switch object to each SIP Server application.
3. On the **Connections** tab, add the following connections:
   - **confserv_proxy_<datacenter>**—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 60
     - Remote Timeout: 90
   - **MessageServer_<datacenter>**—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 7
     - Remote Timeout: 11
4. On the **Server Info** tab, configure the following ports for each SIP Server application:

ID	Listening Port	Connection Protocol
default	Any available port number	
TCport	Any available port number	TController
IPport	Any available port number	IProxy
SmartProxy	Any available port number	SmartProxy
Note: Changes in port numbers take effect after SIP Server restart.		

5. On the **Options** tab in the **[TServer]** section, configure the following mandatory options for each

SIP Server application to be run in cluster mode:

Name	Cluster Value	Description
server-role	5	For SIP Server to run in cluster mode.
sip-link-type	3	For SIP Server to run in multi-threaded mode.
geo-location	<string>	A string identifying the data center to which this SIP Server instance belongs. It is used by SIP Proxy to select SIP Server in same data center as SIP Proxy. SIP Server uses it to select geo-location for 3PCC calls. All applications deployed in the same data center use the same value for this geo-location parameter.
sip-address	<SIP Server A-Record FQDN>	For SIP Server to build the Via and Contact headers in SIP messages.
sip-address-srv	<blank>	For SIP Server to build the Via and Contact headers in SIP messages.
sip-outbound-proxy	true	
sip-enable-gdns	true	To resolve SRV contacts.
sip-enable-rfc3263	true	To resolve priority and weight in SRV tables.
dial-plan	<dial plan DN name>_<short data center name>	The name of the Dial Plan DN (the VoIP Service DN with **service-type** set to `feature-server`).
find-trunk-by-location	true	To enable selection of the trunk and softswitch by geo-location. This is required to keep SIP signalling on the correct data center.
enable-strict-location-match	all	To enable strict matching of MSML resources, which is required for the SIP Cluster. SIP Server in a particular geo-location must only use MCP resources in the same geo-location.
sip-enable-x-genesys-route	true	To enable a private X-Genesys-Route header in SIP messages towards SIP Proxy. It's exclusively used by (and not propagated beyond) the SIP Proxy.
sip-port	<SIP port>	
http-port	<HTTP port>	
management-port	<management port>	

The sample configuration:

```
[agent-reservation]
request-collection-time=300 msec

[backup-sync]
addp-remote-timeout=11
addp-timeout=7
addp-trace=full
protocol=addp

[call-cleanup]
cleanup-idle-tout=60 min
notify-idle-tout=5 min
periodic-check-tout=10 min
```

```
[extrouter]
cast-type=route direct-notoken direct-callid reroute direct-uui direct-ani dnis-pool direct-
digits pullback route-uui direct-network-callid

[Log]
all=/mnt/log/SIPS_<datacenter>_1/SIPS_<datacenter>_1
buffering=false
expire=15
segment=100 MB
spool=/mnt/log/SIPS_<datacenter>_1
standard=network
time-format=iso8601
verbose=all
x-gsipstack-trace-level=3
x-server-trace-level=3

[log-filter]
default-filter-type=hide

[TServer]
acw-persistent-reasons=false
after-routing-timeout=18
agent-emu-login-on-call=true
agent-logout-on-unreg=true
agent-no-answer-action=notready
agent-no-answer-timeout=12
call-observer-with-hold=true
consult-user-data=inherited
clamp-dtmf-allowed=true
default-dn=default_rp
default-route-point=reject=404
default-route-point-order=after-dial-plan
default-music=music/on_hold_saas
dial-plan=DialPlan
divert-on-ringing=false
emulated-login-state=not-ready
extn-no-answer-timeout=12
greeting-call-type-filter=
greeting-delay-events=false
greeting-notification=
http-port=9096
init-dnis-by-ruri=true
logout-on-out-of-service=true
management-port=5002
merged-user-data=merged-over-main
monitor-consult-calls=true
msml-record-metadata-support=true
msml-record-support=true
msml-support=true
music-in-conference-file=qtmf://music/silence
override-to-on-divert=true
posn-no-answer-timeout=12
record-consult-calls=true
record-moh=false
recording-failure-alarm-timeout=900
recording-filename=$UUID$_$DATE$_$TIME$
registrar-default-timeout=140
ring-tone=qtmf://music/ring_back
rq-expire-tmout=0
rq-expire-tout=0
server-id=
set-notready-on-busy=true
```

```
shutdown-sip-reject-code=503
sip-address=<A-record FQDN>
sip-address-srv=
sip-call-retain-timeout=1
sip-dtmf-send-rtp=true
sip-enable-100rel=false
sip-enable-call-info=true
sip-enable-ivr-metadata=true
sip-enable-moh=true
sip-enable-rfc3263=true
sip-invite-treatment-timeout=15
sip-port=5060
sip-preserve-contact=true
sip-treatments-continuous=true
timeguard-reduction=1000
unknown-gateway-reject-code=503
userdata-map-trans-prefix=X-Genesys-
```

Configuring SIP Servers for Virtual Queues

Virtual Queue (VQ) SIP Servers are used primarily to manage Virtual Queues. This eliminates the need to synchronize Virtual Queue states across SIP Cluster Nodes.

1. Deploy SIP Servers as an HA pair (one HA pair per data center), Hot Standby redundancy mode, by following the standard procedure.
   - Suggested application names: **SIPS_VQ_<datacenter>**, **SIPS_VQ_<datacenter>_B**.
2. On the **Connections** tab, add the following connections:
   - **confserv_proxy_<datacenter>**—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 60
     - Remote Timeout: 90
   - **MessageServer_<datacenter>**—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 7
     - Remote Timeout: 11
3. On the **Switches** tab, add the **VQ-switch** object to each VQ SIP Server application. All VQ SIP Servers must be associated with the same **VQ-switch**.
4. On the **Server Info** tab, must be only the default port.
5. VQ SIP Server sample configuration:

```
[agent-reservation]
request-collection-time=300 msec

[backup-sync]
addp-remote-timeout=11
addp-timeout=7
addp-trace=full
protocol=addp

[call-cleanup]
cleanup-idle-tout=60 min
notify-idle-tout=5 min
periodic-check-tout=10 min

[extrouter]
cast-type=route direct-notoken direct-callid reroute direct-uui direct-ani dnis-pool direct-
digits pullback route-uui direct-network-callid

[Log]
all=/mnt/log/SIPS_VQ_<datacenter>/SIPS_VQ_<datacenter>
buffering=false
expire=15
segment=100 MB
spool=/mnt/log/SIPS_VQ_<datacenter>
standard=network
time-format=iso8601
verbose=all
x-gsipstack-trace-level=3
x-server-trace-level=3

[TServer]
acw-persistent-reasons=false
after-routing-timeout=18
agent-emu-login-on-call=true
agent-logout-on-unreg=true
agent-no-answer-action=notready
agent-no-answer-timeout=12
call-observer-with-hold=true
consult-user-data=inherited
clamp-dtmf-allowed=true
default-dn=default_rp
default-route-point=reject=404
default-route-point-order=after-dial-plan
default-music=music/on_hold_saas
dial-plan=DialPlan
divert-on-ringing=false
emulated-login-state=not-ready
extn-no-answer-timeout=12
greeting-call-type-filter=
greeting-delay-events=false
greeting-notification=
http-port=9096
init-dnis-by-ruri=true
logout-on-out-of-service=true
management-port=5002
merged-user-data=merged-over-main
monitor-consult-calls=true
msml-record-metadata-support=true
msml-record-support=true
msml-support=true
music-in-conference-file=qtmf://music/silence
override-to-on-divert=true
posn-no-answer-timeout=12
```

```
record-consult-calls=true
record-moh=false
recording-failure-alarm-timeout=900
recording-filename=$UUID$_$DATE$_$TIME$
registrar-default-timeout=140
ring-tone=qtmf://music/ring_back
rq-expire-tmout=0
rq-expire-tout=0
server-id=
set-notready-on-busy=true
shutdown-sip-reject-code=503
sip-address=<A-record FQDN>
sip-address-srv=
sip-call-retain-timeout=1
sip-dtmf-send-rtp=true
sip-enable-100rel=false
sip-enable-call-info=true
sip-enable-moh=true
sip-enable-rfc3263=true
sip-invite-treatment-timeout=15
sip-port=5060
sip-preserve-contact=true
sip-treatments-continuous=true
timeguard-reduction=1000
userdata-map-trans-prefix=X-Genesys-
```

You will add VQ SIP Servers to the following applications:

- Stat Servers in each data center
- A dedicated HA pair of Interaction Concentrator instances to monitor an HA pair of VQ SIP Servers in the same data center
- Each URS and ORS located in the same data center

Configuring SIP Servers for Historical Reporting

When operating in cluster mode, Interaction Concentrator server (ICON) must connect to two ports of SIP Server: T-Controller (TCport) and Interaction Proxy (IPport). For this purpose, a dummy SIP Server application must be created. When configuring ICON for Voice details, add a connection to the **IPport** of the actual SIP Server application, and add a connection to the **TCport** of the dummy SIP Server application. Each connection represents a session. Genesys Info Mart requires each session to be associated with a SIP Server application.

1. Deploy a SIP Server application, by following the standard procedure.
2. On the **Switches** tab, add the SIP Cluster Switch, the same Switch as in the actual SIP Server application.
3. On the **Server Info** tab, add the same listening ports as in the actual SIP Server application. The Server Info tab must not contain HA configuration.
4. On the **Connections** tab, don't add anything. It must be empty.

5. On the **Options** tab in the **[TServer]** section, don't make any changes.
6. On the **Start Info** tab, clear the Auto-Restart box to avoid SCS restarting the application.
7. On the **Annex** tab in the **[sml]** section, set **autostart**=`false` to avoid SCS restarting the application.

> **Important**
>
> The dummy SIP Server application must *not* be added to the **applications** option of the SIP Cluster DN.

Configuring SIP Proxy

SIP Proxy is an application with active-active HA mode. There should be two SIP Proxy applications deployed in one data center.

1. Deploy two SIP Proxy applications by following the standard procedure. When creating a SIP Proxy application, use the *Genesys Generic Server* type.
   - Suggested application names: **SIPProxy_<datacenter>_1**, **SIPProxy_<datacenter>_2**.
2. On the **Connections** tab, add the following connections:
   - **confserv_proxy_<datacenter>**—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 60
     - Remote Timeout: 90

     SIP Proxy must be connected to the Configuration Server Proxy deployed on the first SIP Cluster Node in the local data center.
   - **MessageServer_<datacenter>**—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 7
     - Remote Timeout: 11

     SIP Proxy must be connected to the Message Server deployed on the first SIP Cluster Node in the local data center.
3. On the **Tenants** tab, add the tenant to be served by the SIP Proxy application.
4. On the **Server Info** tab, set the following parameters:
   - **Host**—Specify the host on which this SIP Proxy is installed.
   - Ports—Specify the following SIP Proxy ports:

ID	Listening Port	Connection Protocol
default	Any available port number	
mgmt-port	Any available port number	mgmt
http-port	Any available port number	http
sip-port	Any available port number	sip

5. On the **Options** tab, create a section named **sipproxy**. In the **[sipproxy]** section, add the following options:

   - geo-location=<the name of the data center>

- oos-check=5
- oos-force=5
- registrar-default-timeout=140

6. Add created SIP Proxy applications to the **sipproxy-applications** option of the SIP Cluster DN.

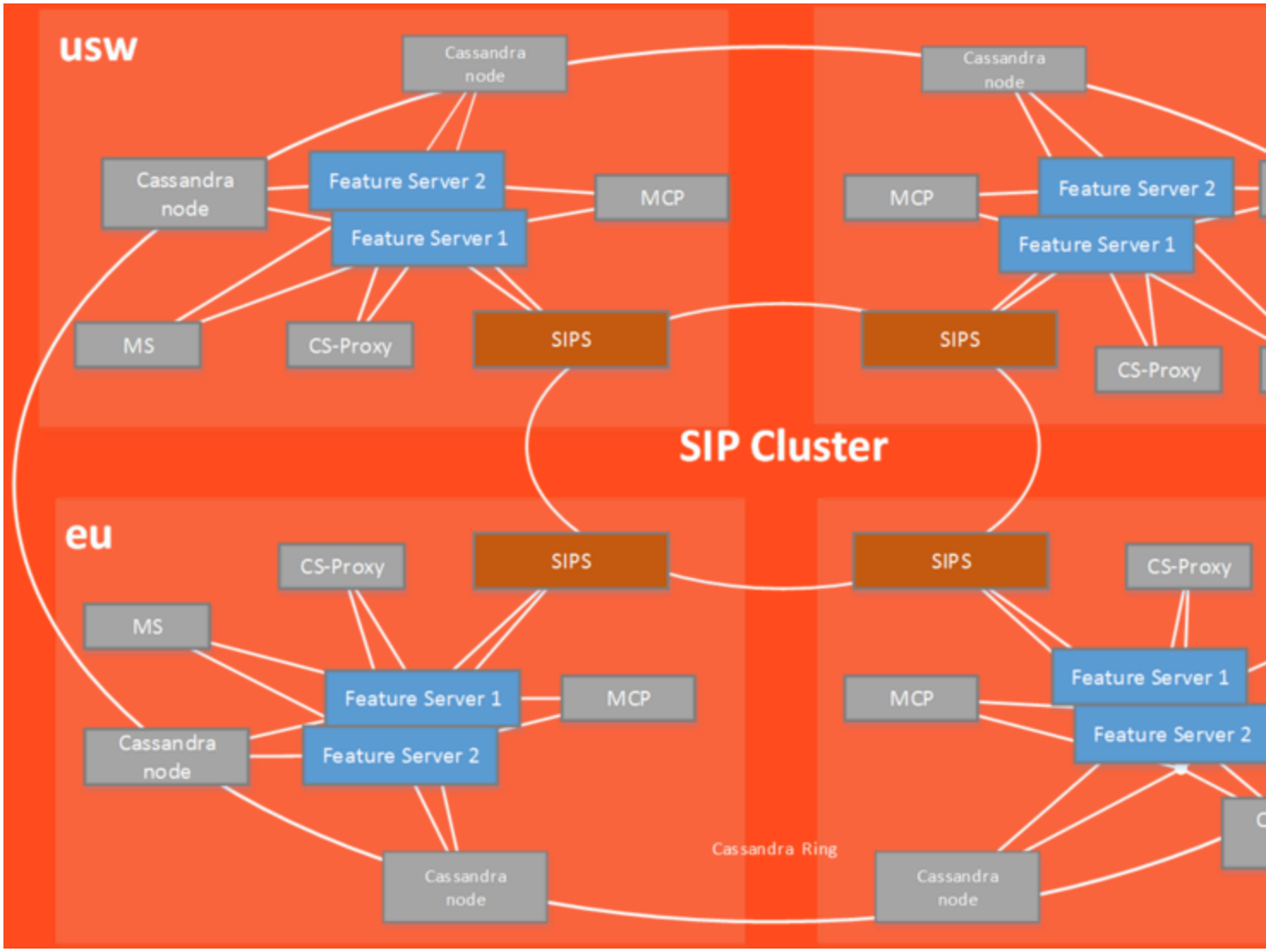
See required provisioning and examples of SRV records in Configuring DNS Records.

Configuring SIP Feature Server

Genesys SIP Feature Server integrates with Genesys SIP Cluster to provide a SIP-based voicemail and SIP feature manager for Genesys contact centers and enterprise environments. Callers leave voicemail, and users retrieve and manage that voicemail. Administrators manage users, devices, voicemail, and call disposition (the dial plan). A distributed architecture enables resiliency and enhances performance.

Standard Feature Server deployment includes a High Availability (HA) installation with the SIP Server in cluster mode and at least one active-active pair of Feature Servers should be deployed to serve one SIP Cluster node. Refer to Deployment options for more details.

You can deploy SIP Feature Server across multiple data centers, in an active-active configuration. The following image shows Feature Server deployment in External Cassandra cluster mode.



Feature Server in SIP Cluster Environment

Configuration and provisioning

Initial program configuration occurs primarily in Genesys Administrator (GA). You can create users, DNs, and mailboxes only in GA, but can use the Feature Server plug-in for Genesys Administrator Extension (GAX) to provision them with devices, voicemail, and call settings.

Dial plan

SIP Feature Server enables the configuration of the dial plan, a highly configurable set of call disposition patterns. You configure the dial plan using either of two methods:

- the Feature Server plug-in for Genesys Administrator Extension (GAX)
- the existing SIP Server methodology, using GA

Voicemail

SIP Feature Server combines with Genesys Voice Platform (GVP) and SIP Server to handle voicemail tasks.

Agents and other users can use the Feature Server GAX plug-in and the Telephone User Interface (TUI) to manage their personal and group voice mailboxes and call settings.

Data management

SIP Feature Server uses Apache Cassandra data clusters to replicate data across the environment, achieving resiliency and high availability. Refer to Data management for more details. It is recommended to use External Cassandra cluster in the SIP Cluster environment.

Feature Server configuration options

See Configuration options for information about the SIP Feature Server configuration options.

Deploying SIP Feature Server

This section explains how to install and configure the SIP Feature Server instances for SIP Cluster.

Planning and pre-installation

Before installing and configuring SIP Feature Server, you must plan your environment and install required hardware and software. In your planning, you must:

- Meet hardware and software prerequisites such as operating system, hardware, and Genesys and third-party components.
- Review the current known issues and recommendations.

Deployment steps

Complete these steps to install and configure the SIP Feature Server instances for SIP Cluster.

- Configure SIP Feature Server applications

> **Important**
>
> For deployments using employee ID as agent login code, set the `use-employee-login` option in the **[VoicemailServer]** section to **TController** on all Feature Server Application objects.

- Deploy a co-located/external Cassandra cluster

- Configure SIP Feature Server for a co-located/external Cassandra cluster

> **Important**
> It is recommended to use External Cassandra cluster.

- Start and verify SIP Feature Server
- Configure SIP Server for Feature Server
- Configure voicemail
- Configure Message Waiting Indicator (MWI)
- Provision users
- Provision DNs
- Provision mailboxes

Configure voicemail

To configure voicemail in Cluster mode:

1. In Genesys Administrator, navigate to the `rm` section of the `Options` tab of the Resource Manager (RM) application, and set the option value of the `sip-header-for-dnis` option to `request-uri`.
2. Create a resource group of type `Media control platform` to make the Media Control Platform (MCP) instances in the cluster available for RM instances.
3. Create a resource group of type `gateway` between SIP Server and each RM instance in the cluster.
4. Under the SIP Cluster Switch, create a unique DN of type **Voice over IP Service** with the same name as the configured Direct Inward Dialing (DID). Create two DNs (DID) to Feature Server HA pair. The voicemail DNs will be used for configuring voicemail.
   - In the Annex > **TServer** section, configure the following options:
     - **contact** = `::msml`
     - **geo-location** = <A string identifying the data center to which this DN belongs>
     - **service-type** = `voicemail`
5. Each SIP node will be connected to one Feature Server HA pair having two voicemail IVR profiles per Feature Server HA pair. Under the Voice Platform tab, create IVR profile for Feature Server instances in the cluster as shown below:
   - `service-type`,value `voicexml`
     For Feature Server 1
   - `initial-page-url`, value `[http: https:]//`*`FQDN1 or //FS1 IP address`*`:`*`port`*`/fs`
     FQDN1 is the FQDN you created while configuring Feature Server applications, if your environment includes more than two Feature Server instances per SIP switch.
   - `alternatevoicexml`,value `[http: https:]//`*`FQDN2 or //FS2 IP address`*`:`*`port`*`/fs`
     FS2 IP address is the IP address of the "extra" Feature Server instance that is not included in *FQDN1*.

For Feature Server 2

- `initial-page-url`, value `[http: https:]//FQDN2 or //FS2 IP address:port/fs`
- `alternatevoicexml`,value `[http: https:]//FQDN1 or //FS1 IP address:port/fs`

6. For each IVR profile created above, configure a unique DID.
7. Configure a GVP DID Group by specifying the following parameter in the Annex tab of the tenant:
   - Under the **[gvp.dn-groups]** section configure the `<DNs in the group>` value for each FS application in the Feature Server HA pair, where `<DNs in the group>` are configured DNs under a Switch with which the voicemail SIP Server Application is associated.

Option	Value
FS1 application name	DID 1
FS2 application name	DID 2

   - Under the **[gvp.dn-group-assignments]** section, configure the `<DBID of the Voicemail IVR Profile>` value for each FS application in the Feature Server HA pair.

Option	Value
FS1 application name	DBID of IVR profile created for Feature Server 1
FS2 application name	DBID of IVR profile created for Feature Server 2

Configure Message Waiting Indicator (MWI)

To configure Feature Server to issue a Message Waiting Indicator (MWI):

1. In your SIP Proxy application, select `Options > sipproxy > feature-server-address`. Configure the Feature Server IP address with port `5160`.
2. To use a different SIP port from the default SIP port on the SIP Feature Server, create a `[sip]` section on the SIP Feature Server and add a **localport** option to assign the SIP port.

Notes:

- The SIP MWI is supported only for individual mailboxes.
- For group mailboxes, only the T-Library MWI is supported.
- Feature Server does not accept subscriptions for device numbers (except where the mailbox number matches the device number).
- To support SIP MWI notification, the SIP endpoints must be configured to subscribe to the voice mailbox number directly.
- The subscription should be sent to SIP Proxy—for example, `mailbox number@SIP Proxy IP`.

Configuring dial-plan in Feature Server

- See Dial plan for information about creating partitions and calling profiles, and editing dial plan settings.

Configuring dial-plan DNs

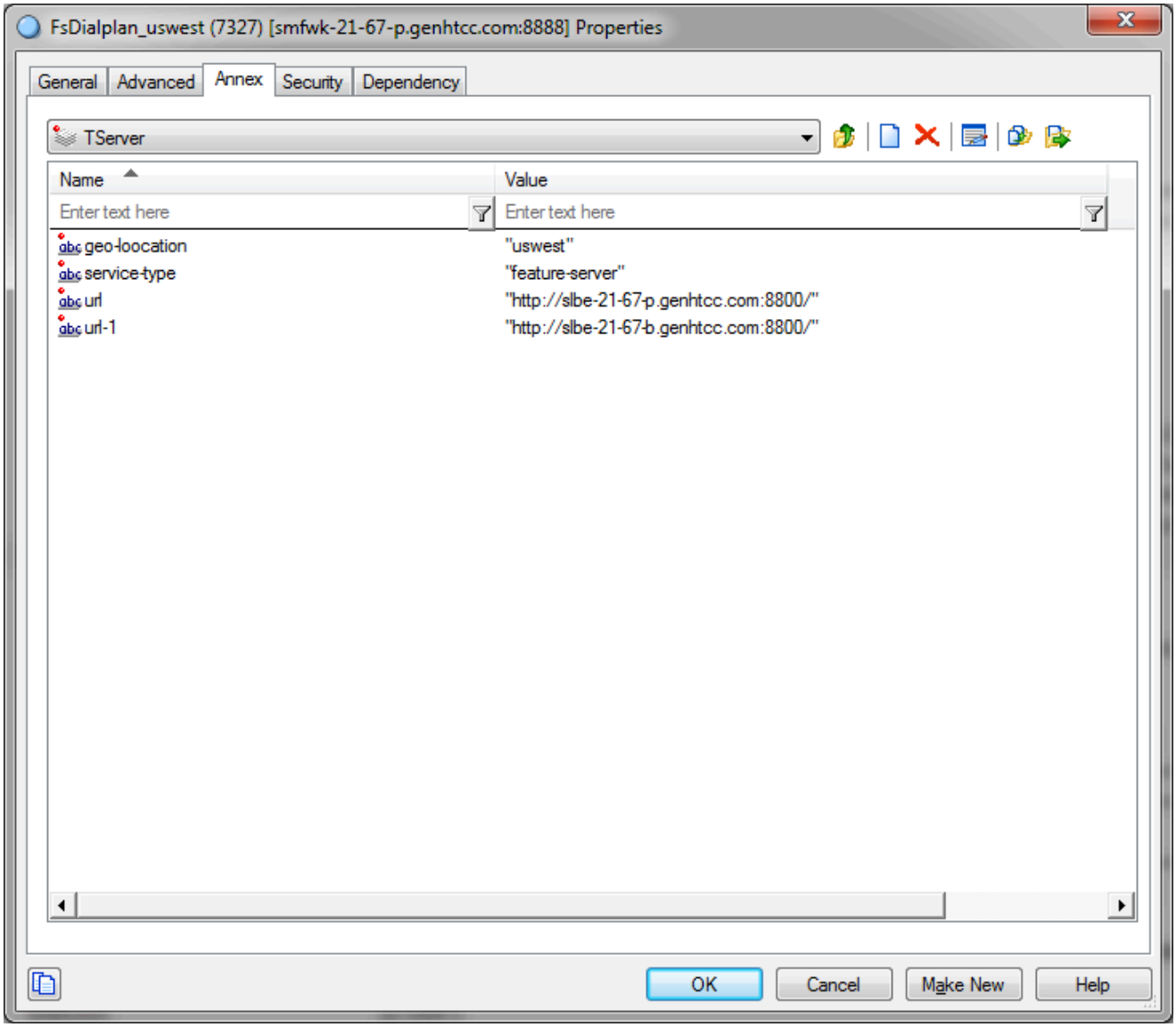
The SIP Cluster Switch might contain one or more dial-plan DNs. A typical SIP Cluster deployment will have one dial-plan DN configured per geo-location.

1. Under the SIP Cluster Switch, create a DN of type **Voice over IP Service** named `dial-plan-<datacenter>`, for each region. For example, **dial-plan-sfo**.
2. In the **Annex** > **TServer** section, configure the following mandatory options:

Name	Value	Notes
geo-location		A string identifying the data center to which this DN belongs.
service-type	feature-server	
url	A URL of a local SIP Feature Server	Set this option to the URL for the local SIP Feature Server in this data center.If there is more than o the data center, create as many **url** options as needed to accommodate all their addresses in one always be named **url**. All consecutive options must be named **url-<*n*>**, depending on the number Server instances (see **url-1** below).
url-1	A URL of the second SIP Feature Server	Set this option to the URL for the second SIP Feature Server in this data center. All consecutive opt **<*n*>**, depending on the number of deployed Feature Server instances.
url-N	A URL of the Nth SIP Feature Server	Set this option to the URL for the N+1th SIP Feature Server in this data center. All consecutive opt **<*n*>**, depending on the number of deployed Feature Server instances.

> **Important**
> For each SIP Feature Server, the pool of connection of URLs depends on the number of SIP Feature Servers available.

Here is a sample screenshot:



Calling profiles and its associations

Calling profile can be assigned to the following contact center objects:

- **Switch**—Generic rules that are applicable for all calls in the cluster deployment.
- **User**—Specific set of rules that need to be configured for a set of users.
- **Extension**—Specific set of rules that need to be configured for a set of extensions.
- **Trunk**—Customized dial-plan for a particular trunk because external calling profile is used for all inbound calls.
- **Softswitch**—Customized dial-plan for a particular softswitch. For example, 10-digit numbers must be converted into E.164 to match PSTN carrier requirement.

- **Trunk Group**—Calling profile used for predictive calls.
- **Route Point**—Calling profile used only for predictive calls. This calling profile does not affect call routing.

Example of minimum required basic dial-plan configuration

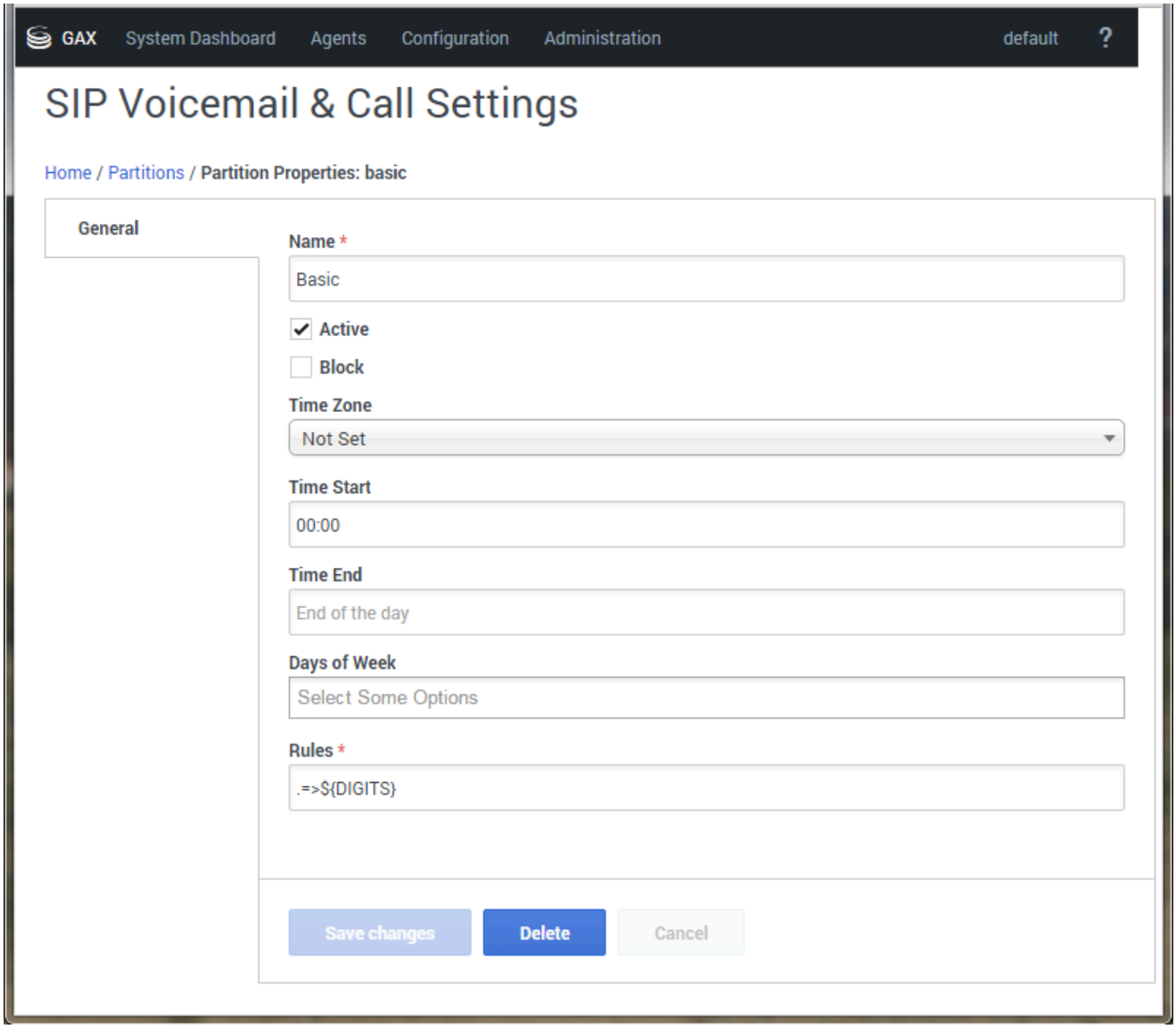
The following table explains the mandatory configuration to use dial-plan.

Name	Description	Rule
Basic	Used for digit translation.	`.=>${DIGITS}`
Voicemail	Used for voicemail access number to `gcti::voicemail` translation.	`5555=>gcti::voicemail`

Partitions

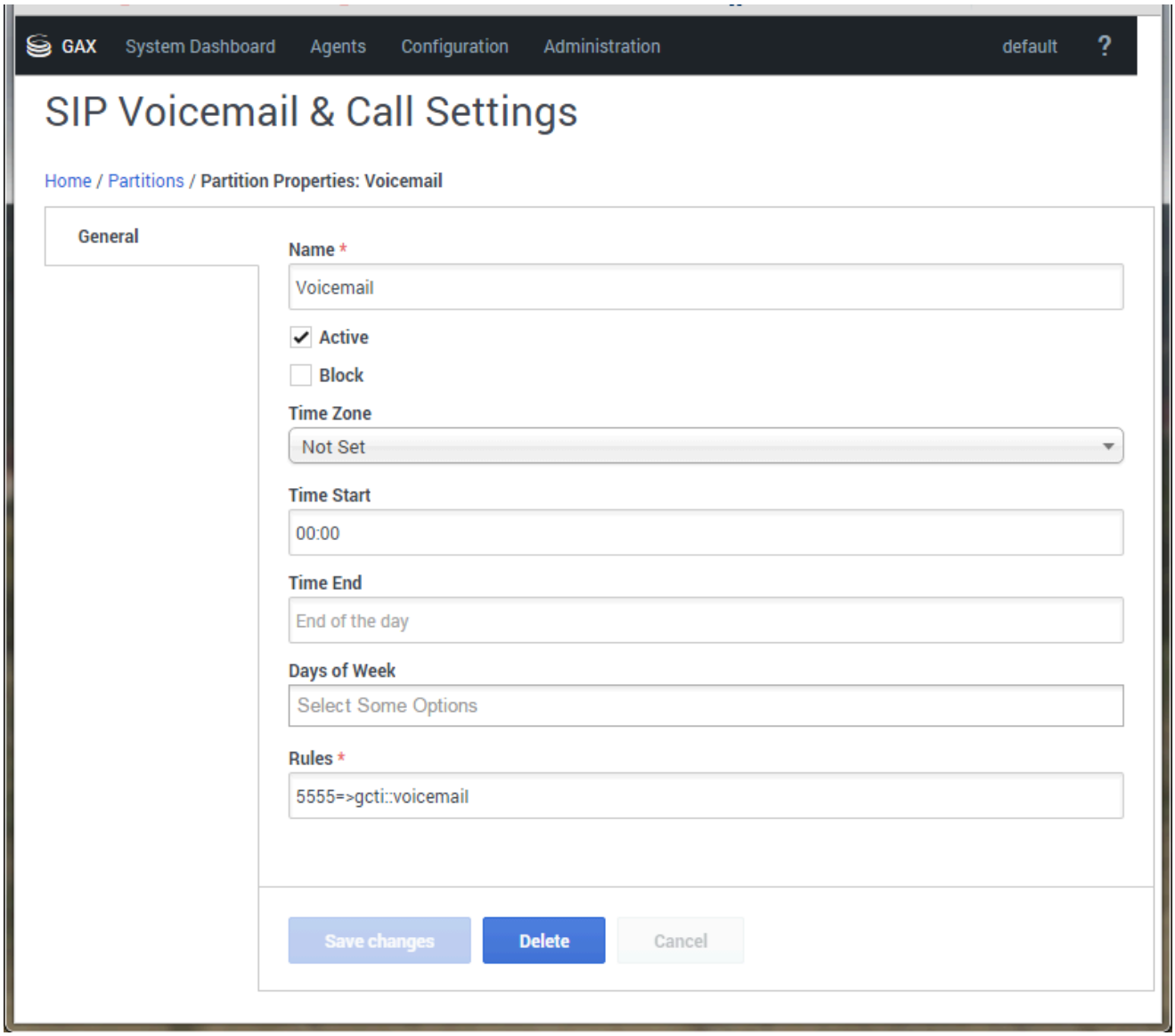
Basic Partitions

"Basic" Partition makes no number translation (Rules: **.=>${DIGITS}**). It is active 24x7.



Voicemail Partition

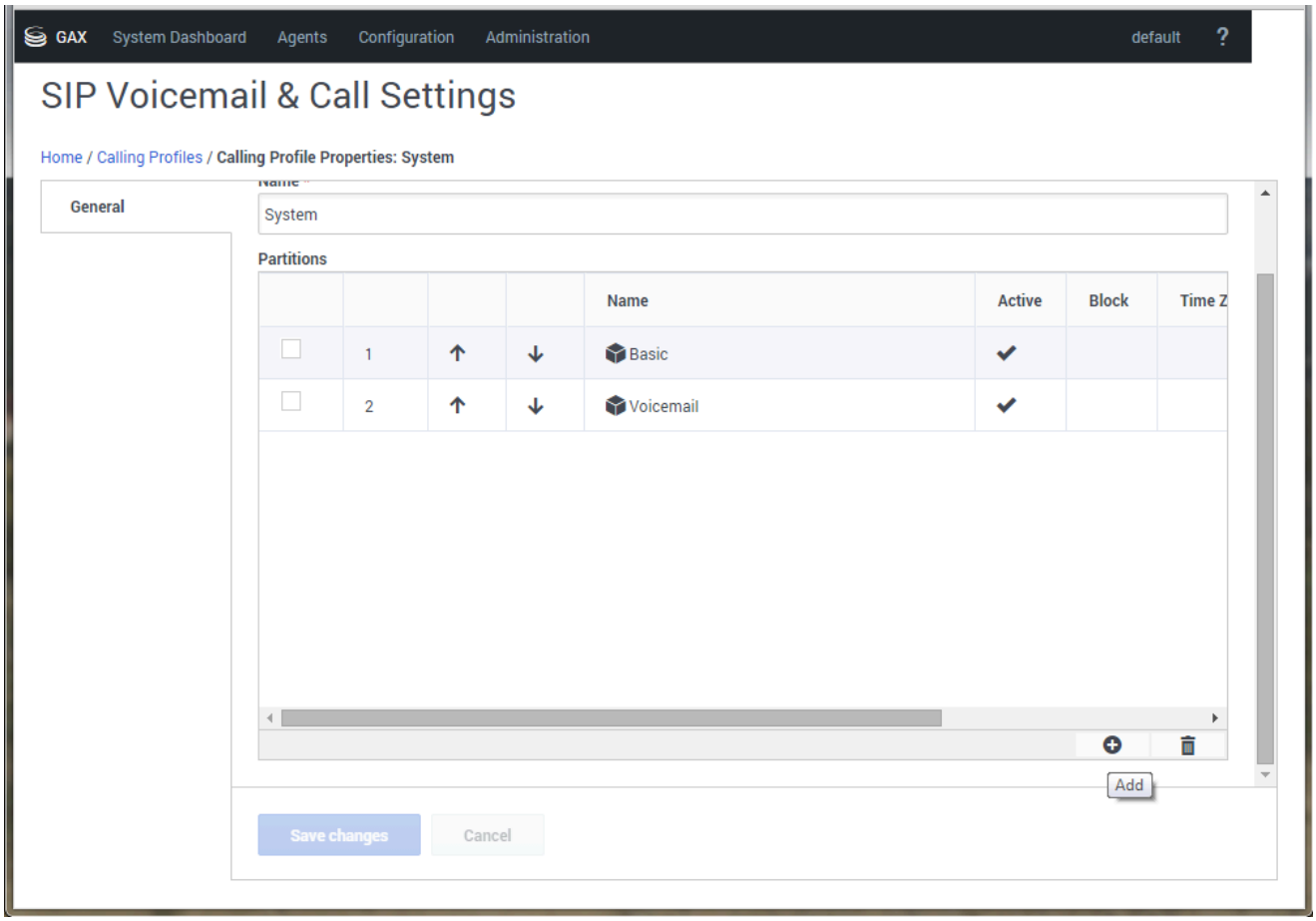
"Voicemail" Partition makes 5555 Voicemail access number (Rules: **5555=>gcti::voicemail**). It is active 24x7.



Calling Profile

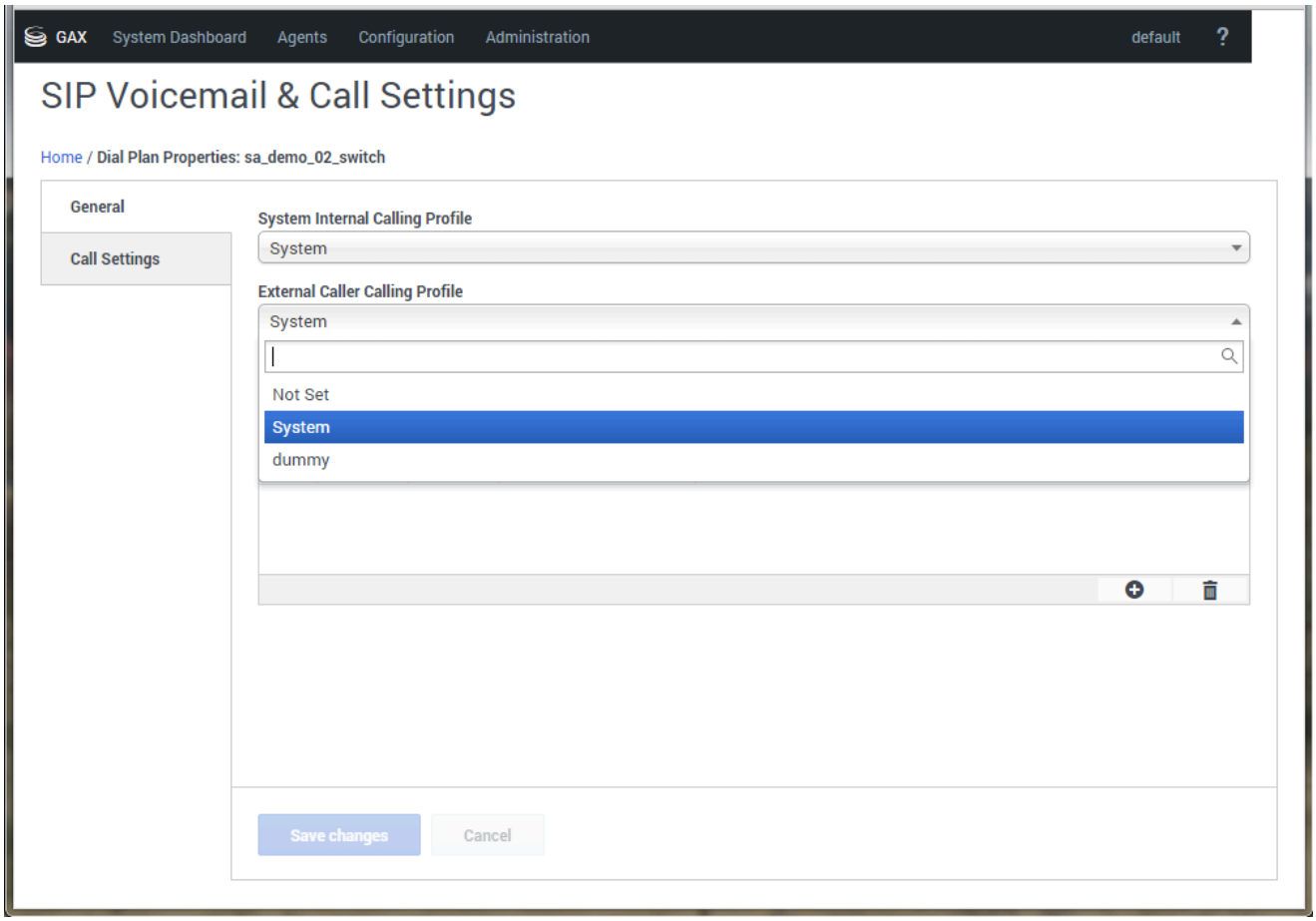
Create Calling Profile

Create Calling Profile System adding Basic and Voicemail Partitions.



Assign Calling Profile

Assign "System" Calling Profile to SIP Cluster Switch as both **System Internal Calling Profile** and **External Caller Calling Profile**.


GAX
System Dashboard
Agents
Configuration
Administration
default
SIP Voicemail & Call Settings
Home / Dial Plan Properties: sa_demo_02_switch
General
Call Settings
System Internal Calling Profile
System
External Caller Calling Profile
System
Not Set
System
dummy
Save changes
Cancel

Configuring GVP

Genesys Voice Platform (GVP) provides the media services for contact center interactions.

Install and configure the GVP solution as described in the GVP 8.5 Deployment Guide.

Configuring Resource Managers

1. Create a Gateway Resource group for tenant identification.
2. Deploy Resource Managers as an active-active HA pair following the standard procedure.
3. On the **Connections** tab, add the following connections:
   - **confserv**—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 60
     - Remote Timeout: 90
   - **MessageServer**—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 7
     - Remote Timeout: 11
   - Add **SIP Proxy** applications to be served by these Resource Managers, using the following connection parameters (usually those SIP Proxy instances are deployed in the same data center with the Resource Managers):
     - logical-resource-section=<GWGroup_NAME>;aor=sip:<SIP-PROXY_A-REC-FQDN>:<SIP-PROXY-PORT>;redundancy-type=active;port-capacity=20000
4. On the **Options** tab, configure the following options in the **[GWGroup_Tenant_<TENANT_ID>]** section:
   - **load-balance-scheme**=`round-robin`
   - **use-cti**=`0`
   - **service-types**=`gateway`
   - **port-usage-type**=`in-and-out`
   - **monitor-method**=`none`

Important

When SIP Cluster is used with GVP deployment, GVP is required to provide media service by using the MCPs deployed in the same Data Center (DC) as the SIP Server that has initiated the request. You must assign a unique geo-location value to each DC. SIP Server will include the custom header “X-Genesys-geo-location” to any INVITE message that requests a new media service.

As a result, with SIP Cluster deployment, you must configure the geo-location parameter for the MCP logical groups (LRG), and provide a list of geo-location values selected for the Data Centers. And this list will be used to configure SIP Cluster and MCP LRGs, accordingly.

Configuring Stat Server

Stat Server tracks information about customer interaction networks (contact center, enterprise-wide, or multi-enterprise telephony and computer networks). Stat Server also converts the data accumulated for directory numbers (DNs), agents, agent groups, and non-telephony-specific object types, such as e-mail and chat sessions, into statistically useful information, and passes these calculations to other software applications that request data.

See the Stat Server Deployment Guide for deploying Stat Server and RTME Options Reference for information about Stat Server configuration options.

Additional Stat Server configuration information

- In a SIP Cluster environment, Genesys recommends to set Stat Server options **[statserver]**/reg-dns-chunk-delay=1 and **[statserver]**/reg-dns-chunk-volume=100.
- When the Smart Proxy module of all SIP Server applications is enabled in a SIP Cluster environment, connect all Stat Server applications to the SmartProxy port of the SIP Server application on the **Connections** tab of Stat Server applications.

Configuring Outbound Contact

Outbound Contact supports a VoIP deployment that enables automated outbound dialing and call-progress detection when using SIP Server in the SIP Cluster environment. Outbound Contact Server (OCS) supports all VoIP dialing modes in the SIP Cluster environment.

Complete the following configuration to enable outbound functionality in your SIP Cluster environment:

- Configure the OCS application
- Configure Switch objects
- Configure a Communication DN
- Configure Trunk Group DNs

Configure the OCS application

1. Follow the standard deployment procedure for deploying OCS.
2. On the **Connections** tab, add the following connections:
   - The primary SIP Server of the SIP Cluster Node that is deployed in the same data center. OCS is connected to the T-Controller port, **TCport**, and only to one SIP Cluster Node in the data center. However, if the Smart Proxy module is enabled, connect OCS to the **SmartProxy** port instead.
   - The primary VQ SIP Server that is deployed in the same data center.
   - The primary Stat Server that is deployed in the same data center.

OCS supports load balancing when operating in a SIP Cluster environment. See OCS Load Balancing in SIP Cluster for details.

Configure Switch objects

In the SIP Cluster Switch and VQ Switch, in the **General** tab, set the **T-Server** field to None.

Configure a Communication DN

1. Under **Switches** > **VQ-switch** > **DNs**, create a Communication DN. Only one Communication DN must be created in the VQ-switch.

2. In the **Annex** tab of the Communication DN, create the **[default]** section. In the **[default]** section, create the **outbound_contact_server** option and set it to `true`.

Configure Trunk Group DNs

1. Under **Switches** > **SIP_Cluster** > **DNs**, create a DN of type **Trunk Group** for each data center.
2. In the **Annex** > **[OCServer]** section, set the **outbound_contact_server** to `true`.
3. In the **Annex** > **[TServer]** section, configure the following configuration options:

Name	Value
contact	::msml
geo-location	<A string identifying the data center to which this DN belongs>
cpd-capability	mediaserver
make-call-rfc3725-flow	1
refer-enabled	false
request-uri	sip:msml@<RM-SRV-FQDN>;gvp-tenant-id=Environment (FQDN of the active-active RM pair resolved in SRV records)
ring-tone-on-make-call	false
subscription-id	Environment

Running Campaign Groups

- When campaign groups run in Active-backup mode, note the following:
  - Only one OCS is used at a time.
  - All campaign groups run on this OCS.
  - OCS in another data center is used only in a disaster recovery scenario.
- When campaign groups run in Active-active mode, they run on the OCS in both data centers.

Limitation: If multiple OCS instances are running campaign groups in parallel in the same environment, one OCS is not aware what campaign groups have been already run on the other OCS.

Configuring Routing

Integrating the Universal Routing Server (URS) and the Orchestration Server (ORS) into the SIP Cluster environment introduces the possibility to distribute the load of calls across several URS and ORS instances. The resources allocated by a given strategy can also be distributed across several URS instances.

URS in a SIP Cluster environment cannot be registered on agents DNs. Therefore, DN-related router functions, IVR treatments, and URS statistics dependent on monitoring agent DNs are not available.

The topics in this section cover important aspects on providing routing services in the SIP Cluster:

- Routing Principles in SIP Cluster
- Agent Availability for Routing
- Agent Reservation in SIP Cluster
- Routing Optimization
- Configuring Orchestration Server
- Configuring Universal Routing Server
- Routing Limitations in SIP Cluster

Related Documentation Resources

The following documentation resources provide general information about URS and ORS:

- Universal Routing 8.1 Deployment Guide
- Universal Routing 8.1 Reference Manual
- Orchestration Server 8.1.4 Deployment Guide
- Orchestration Server 8.1.4 Developer Guide

Routing Principles in SIP Cluster

The following sample deployment diagram illustrates the routing principles in SIP Cluster:

In the above diagram, there are two data centers (DC1 and DC2) with two SIP Cluster nodes in each data center. All SIP Cluster nodes share the same SIP Cluster Switch where all Routing Point DNs and agent DNs are configured. Each SIP Cluster node has a dedicated routing solution consisting of URS, ORS, and Stat Server. A call distributed to a SIP Cluster node is handled by the routing solution of this node, by executing a strategy assigned to a Routing Point.

To simplify provisioning and improve performance, the SIP Cluster architecture includes a switch (VQ Switch) dedicated to Virtual Queue (VQ) DNs. This enables unloading the VQ DN traffic that might be generated in complex large contact center deployments. A single HA pair of SIP Server instances running in non-cluster mode is assigned to that VQ Switch in each data center. For configuration details, see Configuring Switch and DN Objects for SIP Cluster.

Each URS, ORS, and routing Stat Server instance is connected to all VQ SIP Servers in the environment:

- URS - for agent reservation
- ORS - to support GMS/Callback
- Stat Server - to monitor all DNs

Default routing in SIP Cluster

In the SIP Cluster, the call is processed on one SIP Cluster node. When the routing solution serving a node is not operational, SIP Server default routing is applied.

To configure default routing:

1. In the SIP Server Application > Options > **[TServer]** section, configure the following options:
   - **router-timeout**=10 sec
   - **default-dn**=<valid DN>
     - You can specify a Trunk Group DN as the default DN, then a treatment will be played to a call if

both URS and ORS are disconnected from SIP Server.

- You can specify a Routing Point DN as the default DN. So when ORS is disconnected from SIP Server or ORS fails to get the application, URS will execute a default strategy loaded on that Routing Point. Genesys recommends creating a simple strategy for this purpose to be applied to any call.
- You can specify an external number as the default DN for calls to be routed to that specific number.

Next topic: Agent Availability for Routing

Agent Availability for Routing

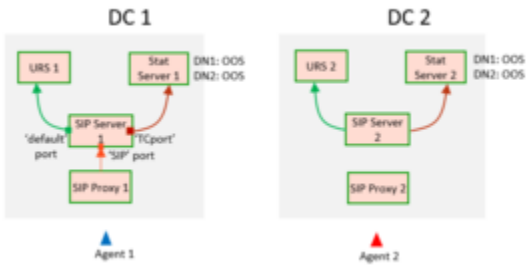
Handling how DNs become available for routing is unique to SIP Cluster.

In the SIP Cluster architecture, URS is connected to the default port of the SIP Server and is only aware of a call on that particular SIP Server. The routing Stat Server is connected to the T-Controller port of the SIP Server through which all calls in the SIP Cluster are passed and, as a result, the Stat Server is aware of all calls on different SIP Servers in the SIP Cluster.

This section discusses DN states and DN ownership based on the type of agent phones: SIP phones or PSTN phones. It also covers the call routing in the SIP Cluster.

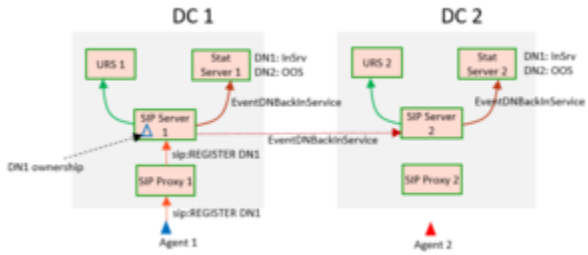
DN State for SIP Phone Agents

When the phones are not registered, the initial state of their DNs is out of service (OOS).



DNs are not registered

The following sample diagram illustrates the flow when one of the SIP phones registers:



One SIP phone is registered

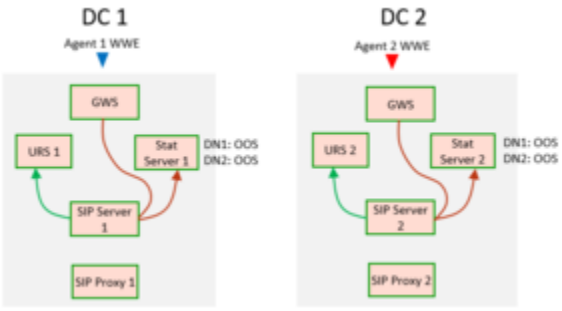
1. SIP phone 1 registers by sending a SIP REGISTER request to data center 1 (DC 1).
2. SIP Server 1 becomes an owner of DN1.
3. DN1 is set to in-service state (InSrv) on SIP Server 1.
4. DN state is passed to the local routing Stat Server through the T-Controller port (TCport).

5. DN state is passed to the remote SIP Server 2 through the T-Controller layer.
6. SIP Server 2 distributes the DN state to the Stat Server 2 through the TCport.

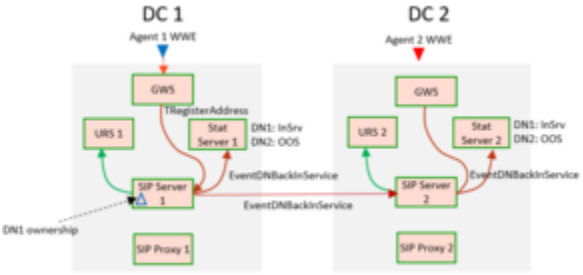
DN State for PSTN Agents

DN ownership for PSTN agents is defined by the T-Library registration, TRegisterAddress request, sent from the Workspace Web Edition (WWE) application into which an agent logs in. The SIP Cluster node received the registration becomes the owner of that DN.

If WWE is not connected, Genesys Web Services (GWS) does not register agent DNs and the initial state of their DNs is out of service (OOS).



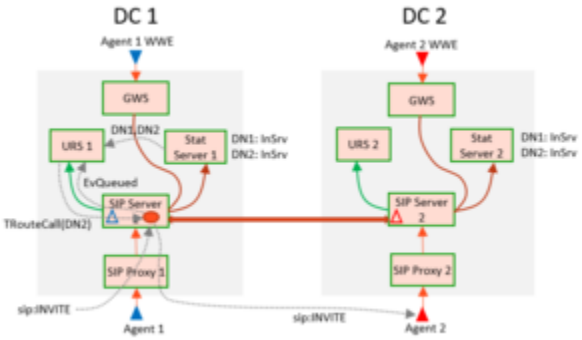
The following diagram illustrates the flow when WWE1 connects to GWS and an agent DN is registered:



1. Agent 1 logs in to WWE and GWS registers on DN1.
2. SIP Server 1 becomes an owner of DN1.
3. SIP Server 1 distributes EventDNBackInService to the local Stat Server and to SIP Server 2.
4. DN1 is set to in-service state (InSrv) on SIP Server 1 and SIP Server 2.
5. DN state is passed to the local routing Stat Server through the T-Controller port (TCport).
6. DN state is passed to the remote SIP Server 2 through the T-Controller layer.
7. SIP Server 2 distributes the DN state to the Stat Server 2 through the TCport.

DN Availability for Routing

The following diagram illustrates how an inbound call is routed to an agent when the SIP Cluster nodes are connected and both DNs (DN1 and DN2) are registered. The call from data center 1 is routed to DN2 on data center 2. A call can be routed from any node to any agent.



The following diagram illustrates the flow when the connection between two SIP Cluster nodes is lost:

As soon as the connection is lost, the DN owner is no longer connected and events for that particular DN cannot be received. SIP Server 1 places DN2 owned by SIP Server 2 into out of service, and vice versa, SIP Server 2 places DN1 owned by SIP Server 1 into out of service. Each SIP Cluster node continues routing to the locally owned DNs.

Next topic: Agent Reservation in SIP Cluster

Agent Reservation in SIP Cluster

Agent Reservation is a very important concept in SIP Cluster. The Agent Reservation flow is illustrated with the following sample diagram:

In the above diagram, Interaction Server handles multimedia interactions for agents. Interaction Server is connected to all routing Stat Servers and reports agent states to all routing Stat Servers in the environment. All routing Stat Servers are aware of the activities of the multimedia agents. Interaction Server is connected to all URS instances in the SIP Cluster and this allows Interaction Server to load balance the routing requests across all available URS instances.

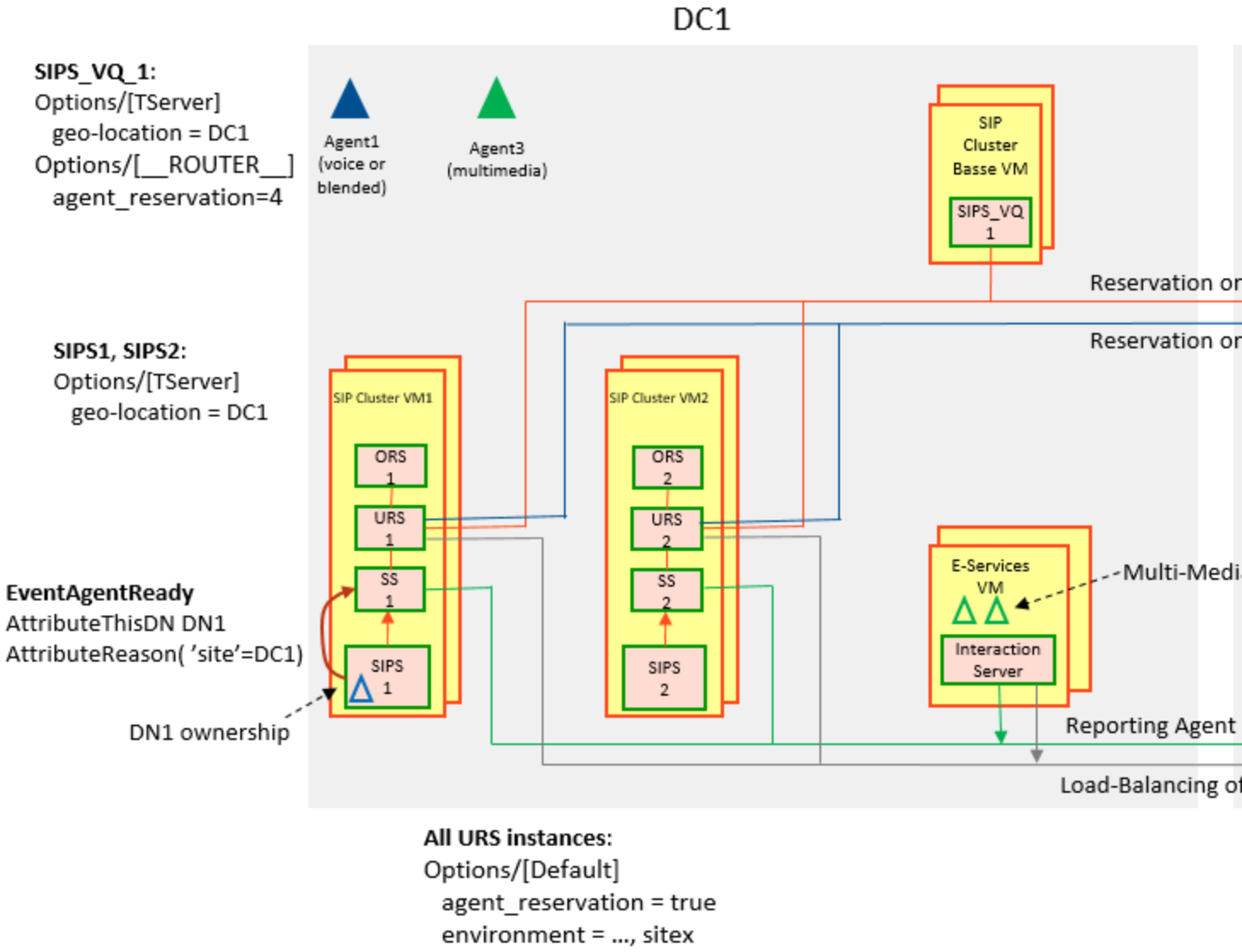
Voice calls arriving to the SIP Cluster node are processed by routing components installed on this node. Agents handling voice calls are logged into the SIP Cluster and their locations are reported to all routing Stat Servers.

A pair of VQ SIP Server instances runs in regular non-cluster mode and is used for agent reservation. URS selects a VQ SIP Server for sending a TRegisterAgent request based on the following configuration:

- If routing is requested to local agents in the routing application (the **environment** option is set in URS), URS selects the VQ SIP Server collocated with the agent, based on the VQ SIP Server's **geo-location** setting.
- If routing is requested without specific agent location or the agent location is unavailable, URS selects the VQ SIP Server based on its priority that is configured in the **agent_reservation** option, the **[__ROUTER__]** section, of the VQ SIP Server.

Reservation Based on Agent Location

The following diagram illustrates reservation based on the agent location:



Agents handling voice calls have DN ownership assigned in the SIP Cluster node. DN ownership defines the agent location. SIP Server reports the agent location in EventAgentReady in the **site** key-value pair of AttributeReason. The value of the **site** key is taken from the application-level **geo-location** option of the VQ SIP Server.

When the **environment** option is explicitly set to `sitex` in URS, URS selects the VQ SIP Server for sending the TReserveAgent request based on VQ SIP Server's **geo-location** setting. In the diagram:

- All requests for Agent 1 reservation go to SIPS_VQ_1, which has **geo-location** set to `DC1`.
- All requests for Agent 2 reservation go to SIPS_VQ_2, which has **geo-location** set to `DC2`.

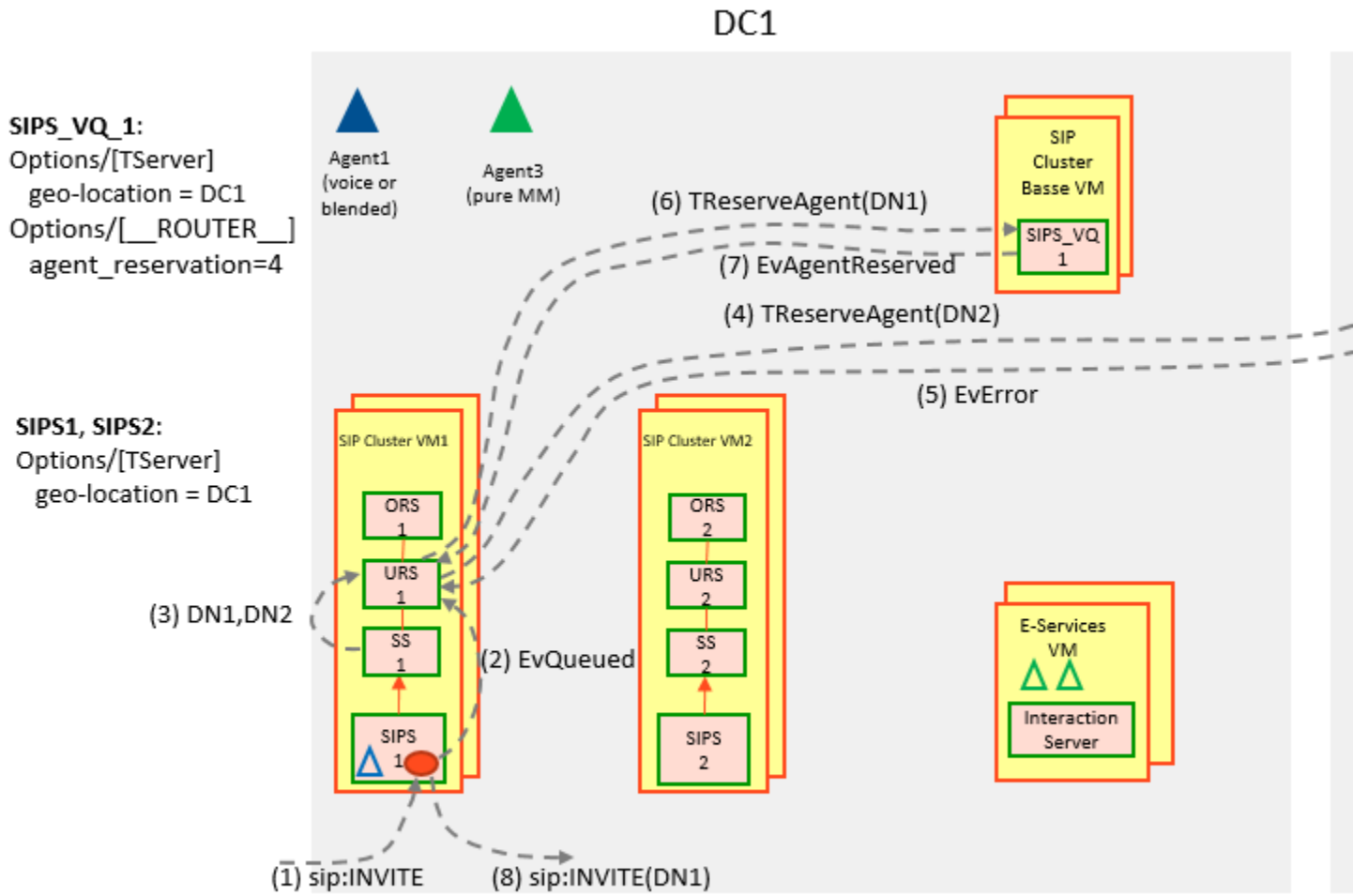
Agents handling multimedia interactions don't have DN ownership assigned. Interaction Server doesn't report the agent location. URS selects the VQ SIP Server for sending the TReserveAgent request based on priority that is specified in the **agent_reservation** option, the **[__ROUTER__]**

section of the VQ SIP Server. In the diagram, the reservation priority of SIPS_VQ_1 is 4, the higher digit the higher priority. So all reservation requests are sent to SIPS_VQ_1. If SIPS_VQ_1 is unavailable, reservation requests are sent to SIPS_VQ_2.

Agent Reservation Conflicts

Agent Reservation conflicts occur when there are multiple URS instances trying to route a call to the same agent.

The following diagram illustrates an example of the reservation conflict:



In the above diagram, there is a call on SIP Server 1 and another on SIP Server 3. The calls are sent to URS 1 and URS 3. The routing Stat Servers on both nodes provide the same set of available DNs (DN1 and DN2). Both URS instances select SIPS_VQ_2 based on the location of Agent 2. The request

from Node 3 is granted and the request from Node 1 is rejected. Node 1 makes another reservation attempt to DN1, which is granted.

Agent reservation conflicts are normal in a SIP Cluster environment where multiple nodes route different calls using the same strategy and target the same group of agents. In the SIP Cluster, conflicts cannot be avoided completely due to the distributive nature of the system, but they can be minimized as discussed in the following section.

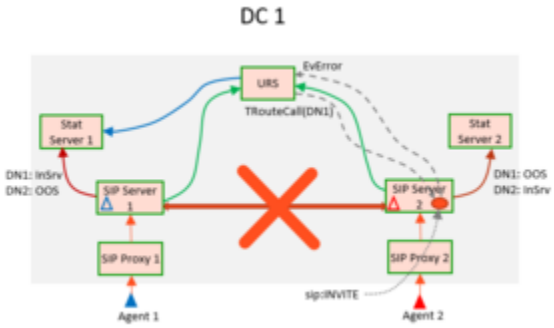
Centralized Routing for Reducing Conflicts

Switching to centralized routing reduces agent reservation conflicts. Consider the following sample architecture diagram:

In the above diagram, there is a centralized URS HA pair that serves the entire data center. There is no separate URS instance serving each node. The fewer instances of URS you use, the lesser the number of reservation conflicts.

URS is connected to the default port that processes local calls from SIP Server 1 and SIP Server 2. URS is also connected to the Stat Server and the Stat Server is connected through the T-Controller layer, which provides URS a global view of all agents and their availability in the SIP Cluster.

However, in a centralized routing scenario, problems might occur when the connection between SIP Cluster nodes is lost and URS no longer has the complete DN availability view. As shown on the following sample diagram, URS is connected to Stat Server 1 and selects DN2 to route a call on Node 2. As a result, URS routes the call to a DN that is out of service, which leads to a failed routing attempt.



Node-Based Routing vs. Centralized Routing

The following table summarizes node-based vs. centralized routing options, their benefits and limitations.

	Node-Based Routing	Centralized Routing
Pros	• Better isolation and modularity (URS maintenance on one node does not affect other nodes) • Better reliability • Better suited for automated deployment	• Fewer agent reservation conflicts (number of conflicts increases with an addition of new data centers or new URS instances in a data center) • Fewer components to deploy
Cons	• Number of agent reservation conflicts grows with the number of nodes	• Less reliable than node-based routing • In case of multiple URS instances per data center, manual pairing of URS and SIP Cluster nodes is required.

Location-Based Routing

Location-based routing is another way to effectively minimize agent reservation conflicts. As discussed in the previous sections, URS has the ability to select a VQ SIP Server based on the agent location using the **site** key-value pair of AttributeReason in EventAgentReady.

Consider a sample scenario where the EventReadyAgent and EventQueued/EventRouteRequest events have the following values:

Location of a ready agent	**Location of a call**
EventAgentReady *AttributeReason* '**site**' '**uswest**' *AttributeThisDN* '7770002' **AgentSessionID** '01M38P5D08AA...' *AttributeAgentWorkMode* 0 *AttributeAgentID* '7770002'	**EventQueued/EventRouteRequest** *AttributeCallType* 2 *AttributeConnID* 0151025a84c2a028 *AttributeDNIS* '5001' *AttributeANI* '8880001' *AttributeThisDN* '5001' *AttributeThisDNRole* 2 *AttributeExtensions* 'OtherTrunkName' 'Trunk_SBC_PSTN_us-west-1' '**geo-location**' '**uswest**'

In the above sample, the value of **site** is populated as *uswest*. When a call arrives at the SIP Cluster node, in the EventQueued/EventRouteRequest messages in the AttributeExtensions, there is a **geo-location** key-value pair. The value of **geo-location** is the same as the **site** key. You can enforce URS to process matching these parameters by creating a skill expression using the **sitex()** function. For example, the routing strategy in Composer can use the following skill expression to route the call to only the agents logged-in to the local data center:

```
sitex(GEO_LOCATION) & English > 0
```

This approach helps to reduce the number of agent reservation conflicts. **geo_location** is passed as a parameter to the **sitex** function. In this case, the reservation requests for agents handling voice calls are load-balanced across data centers and the VQ SIP Server's reservation requests load is reduced.

Composer supports the **sitex()** function. Workflows in Composer provide access to extension data (xdata) to read the geo-location. The Target block accepts a skill expression that can include the **sitex()** function.

Next topic: Routing Optimization

Routing Optimization

Routing can be optimized for higher performance when there is a need for a system where multiple SIP Cluster nodes are required to handle high call volumes. Consider the following sample architecture diagram for a high performance solution:

In the above diagram, there is a central pair of URS instances in each data center. They are connected to extensive ORS clusters serving each SIP Cluster node. In effect, there are more ORS instances than SIP Server instances to process call volumes from each SIP Server instance.

The most efficient optimization is to run ORS on a dedicated virtual machine. You can also increase a number of ORS instances to run on each SIP Cluster node. Reduction of the number of agent reservation conflicts (as in a centralized routing solution) improves performance.

URS Scalability in a Centralized Routing Solution

If one URS pair is not capable of handling the load of the entire data center, you can increase the number of URS instances required per data center and distribute the SIP Cluster nodes to different URS pairs. In the following sample diagram, each URS pair serves two SIP Cluster nodes:

If the capacity of one URS HA pair allows to handle only, for example, two SIP Cluster nodes, new URS instances can be added to the solution as shown above.

Configuring Orchestration Server

1. Deploy ORS applications as an HA pair, Warm Standby redundancy mode, by following the standard procedure. (See the Orchestration Server Deployment Guide.) Each SIP Cluster Node must be served by a dedicated ORS cluster. A dedicated Cassandra ring must be deployed for each ORS cluster.
   - Suggested application names: **ORS_<datacenter>_1**, **ORS_<datacenter>_1_B**.
2. On the **Connections** tab, add the following connections:
   - **confserv_proxy_<datacenter>**—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 60
     - Remote Timeout: 90
   - **MessageServer_<datacenter>**—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 7
     - Remote Timeout: 11
   - **SIPS_<datacenter>_1** (the primary SIP Server, the default port)—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 7
     - Remote Timeout: 11
   - **SIPS_VQ_<datacenter>** (the VQ SIP Server, the default port, located in the same data center as ORS)—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 7
     - Remote Timeout: 11
   - **URS_<datacenter>_1** (the primary URS)—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 7
     - Remote Timeout: 11
3. On the **Options** tab, create the **[gts]** section. In the **[gts]** section, add the **cluster-skip-cfg-event**

option and set it to `0`.

> **Important**
> To create a high performance solution, the number of ORS pairs per SIP Cluster Node can be increased.

Configuring ORS to Reduce URS CPU Usage

URS CPU usage can be reduced to improve performance if required, by:

- Disabling registration of VQ DNs on ORS.
- Having certain SCXML functions to be executed in ORS instead of URS.

The following options must be configured to reduce URS CPU usage:

- **[orchestration]\support-dn-type-5**=`0` (to prevent ORS from registering VQ DNs)
- **[orchestration]\functions-by-urs**=`false` (to enable some SCXML functions to be executed on ORS instead of URS)

Configuring Universal Routing Server

1. Deploy URS applications as an HA pair, Hot Standby redundancy mode, by following the standard procedure.
   - Suggested application names: **URS_<datacenter>_1**, **URS_<datacenter>_1_B**.
2. On the **Connections** tab, add the following connections:
   - **confserv_proxy_<datacenter>**—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 60
     - Remote Timeout: 90
   - **MessageServer_<datacenter>**—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 7
     - Remote Timeout: 11
   - **SIPS_<datacenter>_1** (the primary SIP Server, the default port)—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 7
     - Remote Timeout: 11
   - **SIPS_VQ_<datacenter>** (the VQ SIP Server, the default port, located in the same data center as URS)—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 7
     - Remote Timeout: 11
   - **SS_<datacenter>_1** (the primary Stat Server)—Set to the following parameters:
     - Connection Protocol: addp
     - Trace Mode: Trace On Both Sides
     - Local Timeout: 7
     - Remote Timeout: 11
3. On the **Options** tab, configure the following options in the **[default]** section:

- **default_stat_server**=SS_<datacenter>_1 (the name of the primary Stat Server application for this URS node)
- **agent_reservation**= true

Option descriptions

The **agent_reservation** and **environment** options are configured in the **[default]** section of the URS application.

agent_reservation

Default Value: false
Valid Values: true, false, implicit
Changes Take Effect: Immediately

Enables two or more Universal Routing Servers to work cooperatively by preventing them from trying to route to the same target. To turn agent reservation on, this option must be set to true or implicit for URS. Explicit agent reservation (value set to true) is recommended. Implicit agent reservation doesn't support blended or multimedia agents. In case of explicit agent reservation, this option can be provided at the T-Server level to set T-Server reservation priority.

See Agent Reservation in SIP Cluster and the Universal Routing 8.1 Reference Manual for additional details on this option.

environment

This option is configured at the URS Application-level.

Default Value: None
Valid Values: A comma-separated list of tags
Changes Take Effect: On restart

In case of explicit agent reservation, when this option is set to sitex, URS applies the **geo-location** setting of VQ SIP Servers (in addition to their priority) when selecting the VQ SIP Server to be used for agent reservation. If the **geo-location** is not specified, only the VQ SIP Servers priorities are used for selecting a VQ SIP Server for agent reservation.

For more information on this, see Agent Reservation in SIP Cluster.

Routing Limitations in SIP Cluster

Supported T-Library Requests

The scope of URS within a SIP Cluster environment is to monitor and control Routing Points only. Therefore, the following T-Library requests are supported:

- TRegisterAddress (only for Routing Points)
- TRouteCall
- TApplyTreatment
- TSendDTMF
- TQueryCall
- TQueryServer
- TAttachUserData
- TUpdateUserData
- TDeleteUserData
- TDeleteAllUserData
- TGetAccessNumber
- TCancelReqGetAccessNumber
- TReserverAgent
- TSendEvent
- TPrivateService

IRD Predefined Statistics Not Supported in the SIP Cluster Environment

- RStatLBEWTLAA
- RStatExpectedLBEWTLAA
- RStatExpectedLoadBalance
- RStatLoadBalance

IRD Functions Not Supported in the SIP Cluster Environment

- ACDQ
- AnswerCall
- DeliverCall
- DeliverToIVR
- GetCurrentLeg

- NMTExtractTargets
- ReleaseCall
- ClaimAgentsFromOCS

IRD Objects Not Supported in the SIP Cluster Environment

- Load Balancing

IRD Predefined Macros Not Supported in the SIP Cluster Environment

- MakeAgentNotReady
- RedirectCall
- RedirectCallMakeNotReady

URS Options Not Supported in the SIP Cluster Environment

- transfer_to_agent
- transit_dn
- unix_socket_path
- call_monitoring
- check_call_legs
- count_calls
- lds

URS Capabilities Not Supported in the SIP Cluster Environment

- Cost-based routing
- Support for load balancing
- Support for agents participating in multiple outbound campaigns
- Strategy support for ring-no-answer situations

Configuring Genesys Mobile Services

Genesys Mobile Services (GMS) controls and exposes the Genesys API functionality to external applications by REST APIs. GMS provides a user interface to manage the provisioning and deployment of callback services.

If you are configuring Callback in the SIP Cluster in a multiple URS environment, additional configuration steps are required to ensure that agent reservation conflicts do not occur when a single agent becomes available and there are multiple URS instances trying to route a call to the same agent.

Configure Virtual Queues

In your virtual queues configured for the SIP Cluster, ensure that enough agents will be available by creating, in the **Options** tab, the section **[__ROUTER__]** that contains the **agent_reservation** option.

For example:

```
[__ROUTER__]
agent_reservation=8
```

> **Important**
>
> There are two underscores at the beginning of the section name and two underscores in the ending of section name, that is, 4 underscores in total.

To determine the correct value for agent reservation, see Agent Availability for Routing in this solution guide.

Configure VQ SIP Server

For each VQ SIP Server node of your SIP Cluster, create a **[__ROUTER__]** section that includes **agent_reservation**=`false`.

```
[__ROUTER__]
agent_reservation=false
```

Configure URS

1. Follow instructions for configuring URS as described in Configuring Universal Routing Server.
2. On the **Options** tab, edit the **vcb** option in the **[default]** section:

   ```
   vcb=30:120:60:1:20:0:1:50:1000
   ```

3. Apply changes to the URS backup instance.

Configuring GWS

Web Services and Applications (formerly Workspace Web Edition & Web Services) is a set of REST APIs and user interfaces that provide a web-based client interface to access Genesys services. Web Services and Applications (GWS) supports SIP Cluster with some limitations.

Identification

To identify if the environment is a SIP Cluster, GWS checks if the CloudCluster connections have the **server-role** setting as *5*. Once identified, a **GET ../api/v2/genesys-environments/{id}?fields=*** will return the **clusterMode** flag enabled in the response and the Employee login can be used as the agent login.

```
"voiceEnvironments": [
{
  "backupAddress": "%backup_address%",
  "backupDbid": 371,
  "backupPort": "5001",
  "backupServerName": "SIPS_B",
  "backupSipPort": "5060",
  "clusterMode": true,
  "connectionProtocol": "addp",
  "id": "53ef36a3-e10d-45cc-82dd-16db9a600fb1",
  ...
  "tserverConnections": [
  {
    "backupAddress": "%backup_address%",
    "backupDbid": 371,
    "backupPort": "5001",
    "backupServerName": "SIPS_B",
    "backupSipPort": "5060",
    "connectionProtocol": "addp",
    "genesysEnvironmentId": "2ad2a3b5-5f8a-47b0-b196-7465b959416e",
    "id": "6e35d741-db06-4eae-a95f-743c5ef4fe83",
    ...
  }]
}
...
}
...
```

The **useEmployeeIdAsAgentLoginForSIPCluster** option in the **application.yaml** file must be set to *true*. This setting enables the employee ID to be used as the agent login. The StartContactCenterSession call will use the employee ID as the login ID.

Deployments

In a SIP Cluster deployment, WWE is the agent-facing UI and Genesys Softphone is the software SIP Endpoint. There are two possible deployments that GWS supports for SIP Cluster:

- Standalone mode
- Connector mode

Standalone Mode Deployment

In a Standalone deployment, WWE and Softphone run side-by-side in the same data center. The settings for the Softphone including REGISTER URI (including a DN), are shipped with binaries in a configuration file.

Connector Mode Deployment

In a Connector deployment, WWE configures and controls Softphone during runtime. The settings for Softphone is identical in all instances and the configuration is centralized in the configuration environment (including a DN).

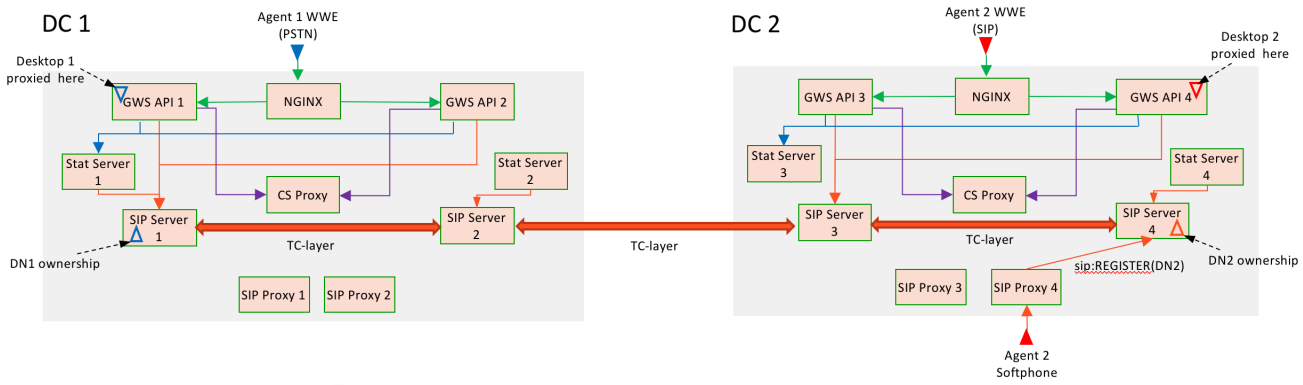
Provisioning

To support SIP Cluster, the following items have to be provisioned.

Person	• For Standalone Mode, the defaultPlace is not required for hot seating. • For Connector Mode, the defaultPlace is required for hot seating. The Employee ID is used as the AttributeAgentID in TAgentLogin requests when the **useEmployeeIdAsAgentLoginForSIPCluster** option is set to *true* in the **application.yaml** file. Extension DNs are configured under the SIP Cluster switch.
Place	The Extension DN must be configured.

The Agent Login objects should not be configured in the SIP Cluster environment.

GWS Connection to 4-node 2-DC SIP Cluster



All GWS API nodes in one data center connects to:

- one SIP Cluster node
- one Stat Server
- Configuration Server Proxy

All connections have the **locations** parameter specified in the **Application Parameters** section of the **Advanced** tab:

- locations=/APS3

See Multiple Data Center Deployment for more information on configuring multiple data centers.

SIP Access Point Configuration

Ensure the following items are configured:

- All SIP traffic to SIP Cluster should be sent to one geo-aware SRV FQDN address, e.g.,sipcluster.abc.com.
- The SIP Cluster access point SRV FQDN is configured in the SIP Outbound Proxy DN of type Voice over IP Service as a value of the **external-contact** parameter in the **[TServer]** section.
- SIP Cluster sends this FQDN as its contact in the outgoing SIP messages. e.g., Record-Route: sipcluster.abc.com
- All SIP clients of the SIP Cluster must support the SRV FQDN and should be able to resolve the SIP Cluster access point SRV FQDN:
  - RM, SBC, Softphones
- DNS records should be configured to resolve the SRV FQDN to the SIP Cluster access point closest to the requestor.
- SIP Cluster access point may be SIP Proxy, or SBC, or both.

WWE Provisioning for Softphone

Configure Genesys Softphone with the following patterns, *privilege.sipendpoint.XXX*:

- privilege.sipendpoint.can-use
- privilege.sipendpoint.can-change-microphone-volume

Control Parameters

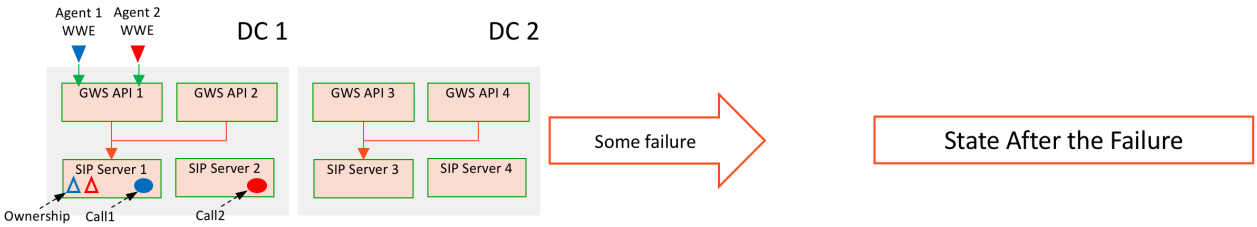
Softphone Control Parameter	• sipendpoint.uri: http[s]://localhost:<softphone-listening-port> Ensure that the protocol is aligned with the GWS browser connection protocol (http or https).
SIP Configuration Parameters	• sipendpoint.sip-server-address: SRV FQDN pointing at the SIP Cluster SIP access point, e.g., sipcluster.abc.com • sipendpoint.transport-protocol: udp/tcp/tls
Softphone Specific Parameters	Based on the pattern: *.<domain>.<section>.<setting>*, all parameters in this category are automatically sent to the Softphone. The Softphone overwrites local configuration file settings with the values received from the connector: • sipendpoint.policy.device.use_headset=1 • sipendpoint.policy.device.headset_name=*

GWS Disaster Recovery Scenarios

> **Important**
> This page lists the several DR Scenarios when using GWS with SIP Cluster.

The following scenario is assumed to be the standard environment setup for the disaster recovery scenarios:

- Two PSTN agents are connected to the GWS1@DC1.
- SIP Cluster DN ownership for the PSTN agents is established on the SIP Cluster node where GWS is connected (SIP Server 1).
- Both agents are logged in.
- Agent1 handles Call1 and Agent2 handles Call2.

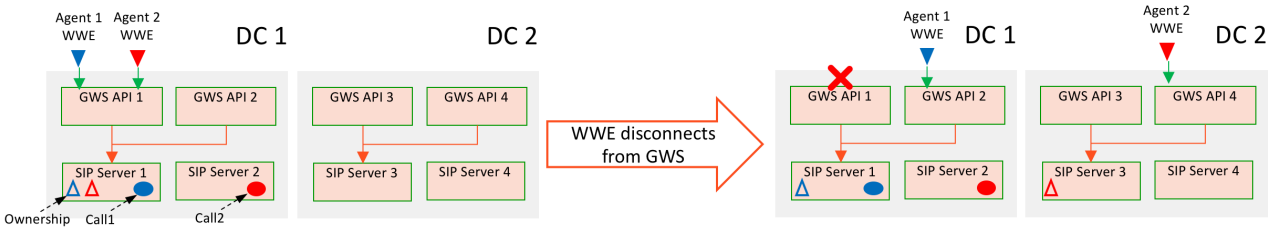


> **Warning**
> If best practices are applied, then failures, which occur in DC2 or with the connection link between DC1 and DC2, do not affect calls and agents in DC1, but local DC1 failures still do.

Some of the sample DR Scenarios covered are:

- WWE reconnects to a new GWS API node
- GWS disconnects from SIP Server
- TC-layer between DC1 and DC2 fails
- Softphone disconnects from DC1

WWE reconnects to a new GWS API node



Failure Condition

- WWE disconnects from GWS1 (e.g., NIC failure).
- SIP and media paths are not impacted.
- Agent can continue talking to the customer.

Summary

- Agents can continue talking to the customers at all times.
- Agents are logged out from their desktop and need to log on again.
- Agent2 may need to log on twice.
- Agents temporarily lose call control through WWE.

Agent 1

WWE Error messages

- “Connection to the server has been lost. Reconnecting…”
- If not recovered after 1 minute, “Unable to establish connection. Please refresh your browser and log in again”

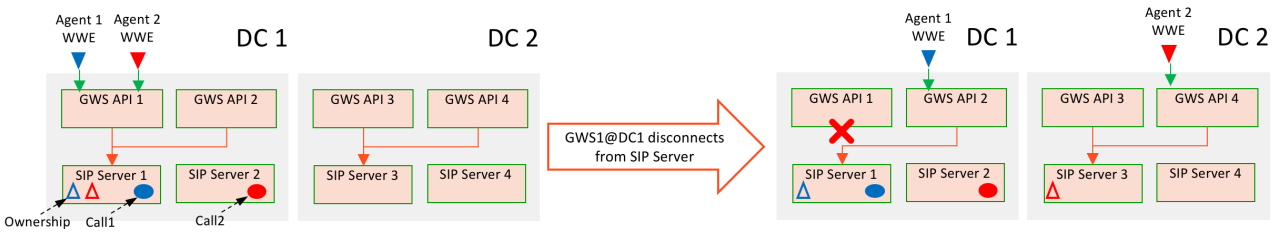
Recovery

1. Agent 1 connects to GWS2@DC1 and NGINX forwards WWE to the other node in the same DC.
   - If GWS1 is down, NGINX drops the sticky session and opens a new session to GWS2.
2. Agent 1 logs on again and GWS2@DC1 submits the following requests:
   - TRegisterAddress(DN1): existing call details are reported.
   - TAgentLogin(DN1): rejected because there is a call in progress.
3. Agent can control the call through WWE but cannot log in. The call is released and ACW is over.
4. Agent logs in to GWS1@DC1 with the TUnregisterAddress(DN1) request.
   - SIP Cluster doesn’t move the ownership because new desktop application connects to the same SIP Cluster node.

Agent 2

1. Agent 2 undergoes the same process as Agent 1 till Step 4 (when GWS1@DC1 submits TUnregisterAddress(DN2)).
   - It happens independently of call or ACW state.
2. SIP Cluster waits until call and ACW is over and moves the ownership to SIP Server 3 @DC2.
3. If Agent 2 logs on successfully before that, then SIP Server logs Agent 2 out before moving the ownership and Agent 2 has to log on again.
4. If GWS1 fails and SIP Server 1 detects it, there is no TUnregisterAddress(DN2) received but SIP Server triggers logout-on-disconnect mechanism, which leads to agent logout for Agent 2.
5. If GWS1@DC1 doesn't submit TUnregisterAddress(DN2) and doesn't disconnect from SIP Server, then ownership is not transferred.

GWS disconnects from SIP Server



Failure Condition

- GWS1 disconnects from SIP Server 1 @ DC1 (e.g. NIC or network failure).
- GWS1 is still running.
- SIP and media paths are not impacted.

Summary

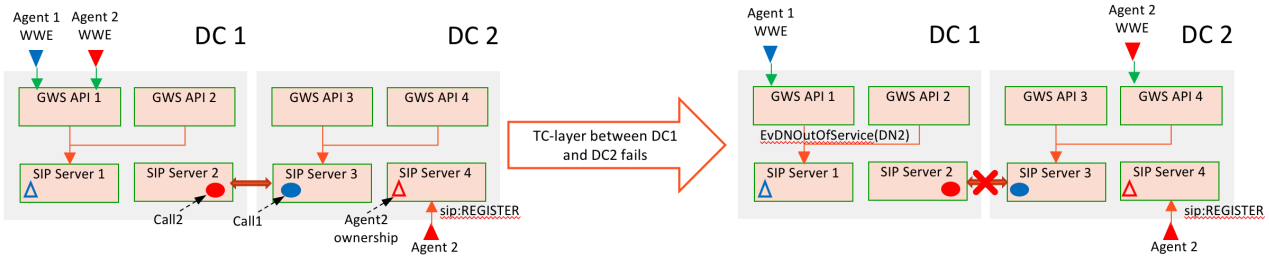
- Agents can continue talking to the customers at all times.
- Agents are logged out from their desktop and need to log on again.
- Agent 2 may need to log on twice.
- Agents temporarily lose call control through WWE.

Details

1. GWS disconnects from SIP Server.
2. SIP Server detects that the desktop client is disconnected.
3. SIP Server logs out an agent (logout-on-disconnect = true), but the call is preserved and agent continues talking to the customer.

4. GWS notifies the agent about the voice channel in OutOfService state and the loss of connection to the SIP Server.
5. WWE reports the voice channel as unavailable.
6. WWE reconnects to the available GWS node when call is still in progress.
7. Agent can control the call but cannot log in.
8. Call is released and ACW is over.
9. Agent logs on.
10. If agent is connected to the new DC, SIP Server waits for 30 seconds, logs the agent out, and moves the ownership.
11. Agent has to log on again.

TC-layer between DC1 and DC2 fails



Conditions

- Agent 1: PSTN agent as in previous example
- Call1: SIP Server 3 @ DC2
- Agent2 is a Softphone agent:
- Ownership: SIP Server 4 @ DC2
- Desktop: GWS1@DC1 connected to SIP Server 1 @ DC1
- Call2: SIP Server 2 @ DC1

Failure Condition

- TC-layer fails between DC1 and DC2.
- SIP and media paths are not impacted.

Summary

- Best practices are not followed.
- Agent1: Call and agent are in different DC's (not a local routing).
- Agent 2: Ownership and desktop are in different DC's (priorities of WWE FQDN and SIP FQDNs are not aligned).

- Agents can continue talking to the customers at all times.
- Agents lose control over the call (for different reasons).
- Agent 2 may need to log on twice.

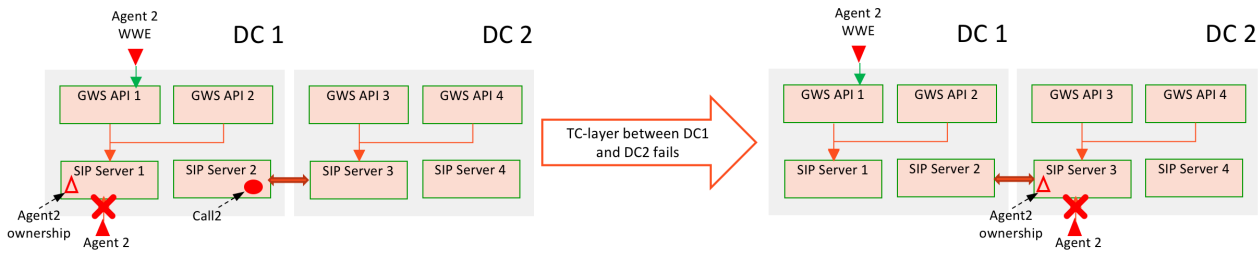
Agent 1

1. Call continues, audio is not impacted.
2. Agent loses 3pcc control over the call and doesn't receive call-related T-Events.
3. As soon as actual call is over, Agent 1 can submit TReleaseCall from WWE to avoid getting a stuck call on a desktop.
4. Call control comes back if TC-layer is recovered.
5. Agent 1 can restore 3pcc control over the call by reconnecting to DC2 but there is no notification suggesting to do that.

Agent 2

1. There is no problem for Agent 2 after the failure as SIP and RTP connections are still OK between the SIP Server 2 @ DC1 and agent's phone.
2. SIP Server 1 @ DC1 has lost connection to the DN owner.
   - Agent 2 is logged out and the DN is set to OutOfService state.
   - Audio channel is not impacted.
3. WWE reports that voice channel is not available.
4. SIP Server 4 @ DC2 logs out the agent based on 'logout-on-disconnect' = true
   - The call continues.
5. Agent 2 can not log on to any GWS node in DC1.
6. Agent 2 should reconnect to DC2 to log on.
7. Agent 2 can log in.
   - Agent 2 can continue talking to the customer.
   - But there is no call reported in the desktop because the call is in the disconnected DC.

Softphone disconnects from DC1



Conditions

- Agent2 is a Softphone agent:
- Ownership: SIP Server 1 @ DC1
- Desktop: GWS1@DC1 connected to SIP Server 1 @ DC1
- Call2: SIP Server 2 @ DC1

Failure Condition

- Softphone cannot connect to DC1 .
- The call is lost because both SIP and media connections are affected at the same time.
- If the connection is restored when existing registration hass not expired, then DN stays in service and agent is not affected.

Details

1. The connection to DC1 is lost permanently.
2. Softphone detects the failure when attempting to refresh the registration and registers at SIP Server 3 @ DC2.
3. There is no call or ACW.
4. SIP Server 1 @ DC1 logs out an agent and immediately moves the ownership.
5. WWE reports *Unable to establish connection. Please refresh your browser and log in again* error message.
6. Agent has to log on again.
7. Agent has to log on to any DC.

Historical Reporting

Historical Reporting Components

The SIP Cluster solution includes the Genesys Historical Reporting products that are briefly described in the following sections:

- Interaction Concentrator
- Genesys Info Mart
- Reporting and Analytics Aggregates
- Genesys CX Insights

Review the information in this Solution Guide to learn how the deployment and operations of these products in the SIP Cluster solution differ from other types of deployments. For complete information about these products, see the product-specific documentation that is described in Related Documentation Resources.

Interaction Concentrator

Interaction Concentrator collects and stores detailed data about the interactions and resources in customer interaction networks that use Genesys Framework. Operating on top of Genesys Framework, the Interaction Concentrator product consists of:

- Interaction Concentrator (ICON) server application, which receives data from data sources such as Configuration Server, SIP Server, or certain other Genesys applications.
- Interaction Database (IDB) in which the ICON server stores the collected data by using Genesys DB Server.

Genesys Info Mart

Genesys Info Mart provides a data mart that you can use for contact center historical reporting. Genesys Info Mart consists of:

- Genesys Info Mart server component, which extracts, transforms, and loads data into a data mart, based on a schedule that is configured in the Genesys Info Mart application.
- Genesys Info Mart Manager (GIM Manager), which provides a graphical user interface (GUI) to manage some of the extract, transform, and load (ETL) processes. GIM Manager is a plugin of Genesys Administrator Extension (GAX) and, consequently, requires GAX in order to run.
- Info Mart database, which stores low-level interaction data that is consolidated from any number of Interaction Databases (IDBs), as well as processed data that is suitable for end-user reports.

Genesys Info Mart can also be configured to host an aggregation engine (RAA) that aggregates or re-

aggregates the data and populates aggregate tables in the Info Mart database.

RAA and GCXI

For optional out-of-box aggregate reporting, you can use:

- Reporting and Analytics Aggregates (RAA), which is an optional Genesys Info Mart process that you can add to an existing Genesys Info Mart environment to create and populate predefined aggregation tables and views within an 8.5 Genesys Info Mart. This aggregation process is essential for GCXI environments.
- Genesys Customer Experience Insights (GCXI), which is the presentation layer for the business-like interpretation of source data that is collected and stored in the Info Mart database. GCXI requires RAA.

Reporting Data Changes

This section highlights the specifics about the reporting data that the Historical Reporting products produce when they operate in the SIP Cluster solution.

Interaction Concentrator

Much of the configuration and functionality for Interaction Concentrator remain the same in a Cluster environment as in a non-Cluster environment. Many of the differences that result from the requirements of a Cluster architecture are entirely transparent to you. The major differences that affect you are of the two following types:

- Differences in the configuration that is required to ensure that Interaction Concentrator functions correctly and produces the results that you expect, as described in Enabling Historical Reporting.
- Differences in the data that is produced in a SIP Cluster environment, as discussed in Historical Reporting Deployment Considerations.

Improvements and differences in ICON operation in a Cluster environment are described in Historical Reporting Operational Considerations.

Genesys Info Mart

The data that Genesys Info Mart produces in a Cluster environment is similar to the data that is produced in a non-Cluster environment, with a few exceptions. The data differences result from differences in resources (agents and DNs) processing in a SIP Cluster deployment.

For more information, see Historical Reporting Deployment Considerations and Historical Reporting Operational Considerations.

Distributed Data Centers

Historical Reporting can be deployed to support business continuity and disaster recovery

requirements in a SIP Cluster environment with two data centers. For an architectural description and deployment recommendations, see Historical Reporting and SIP Business Continuity.

Related Documentation Resources

The information in this *Solution Guide* focuses on aspects of functionality or deployment in which historical reporting in the SIP Cluster solution differs from other types of deployments. For general information about the reporting-related applications that provide historical reporting in the SIP Cluster solution, consult the following documentation resources.

Note: These publicly available documents do not reflect updates for SIP Cluster configuration. The information specific to SIP Cluster deployment is provided in this *Solution Guide.*

Interaction Concentrator Documentation

- Interaction Concentrator Deployment Guide
- Interaction Concentrator User's Guide
- Interaction Concentrator Physical Data Model for a Microsoft SQL Database
- Interaction Concentrator Physical Data Model for an Oracle Database
- Interaction Concentrator Physical Data Model for a PostgreSQL Database
- Interaction Concentrator 8.1 Database Size Estimator

Genesys Info Mart Documentation

- Genesys Info Mart 8.5 Deployment Guide
- Genesys Info Mart Options Reference
- Genesys Info Mart 8.5 Operations Guide
- Genesys Info Mart 8.5 User's Guide
- Genesys Info Mart 8.5 Physical Data Model for a Microsoft SQL Server Database
- Genesys Info Mart 8.5 Physical Data Model for an Oracle Database
- Genesys Info Mart 8.5 Physical Data Model for a PostgreSQL Database
- Genesys Info Mart 8.5 Database Size Estimator
- Genesys Info Mart Manager 8.5 Help
- Genesys Info Mart Manager 8.1 Business Continuity Deployment Guide

Reporting and Analytics Aggregates Documentation

- Reporting and Analytics Aggregates 8.5 Deployment Guide
- Reporting and Analytics Aggregates 8.5 User's Guide

- Reporting and Analytics Aggregates Options Reference
- Reporting and Analytics Aggregates 8.5 Physical Data Model for an Oracle Database

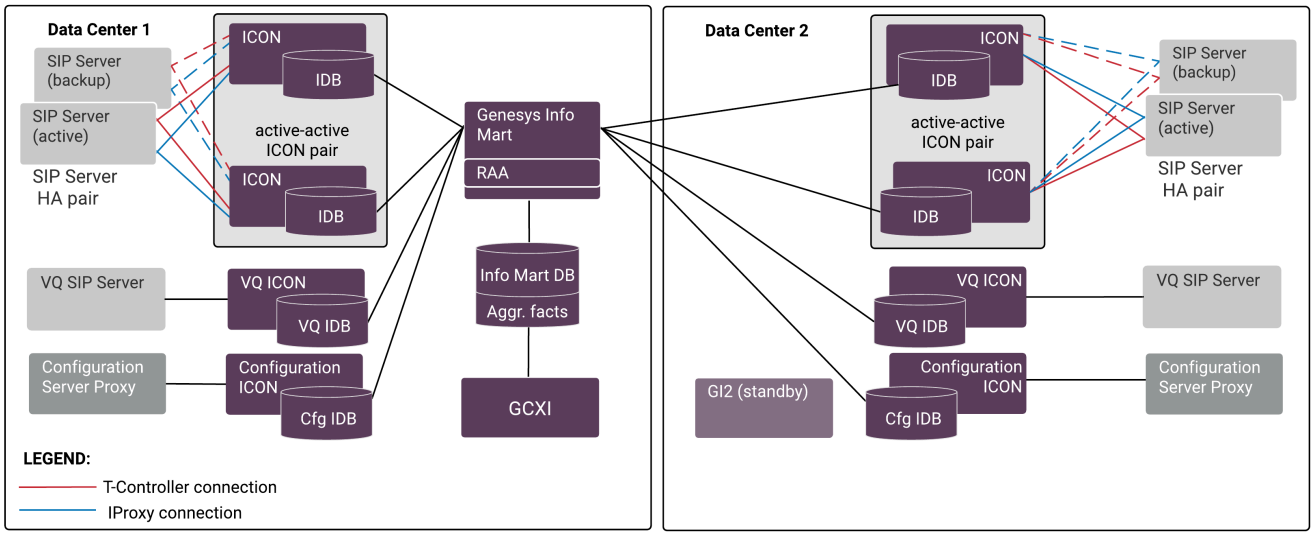
Genesys CX Insights

- Genesys CX Insights Deployment Guide
- Genesys CX Insights User's Guide
- Genesys CX Insights Projects Reference Guide
- Genesys CX Insights Hardware Sizing Guide

Historical Reporting Architecture

Recommended Architecture

To collect historical reporting data from the Genesys applications in a SIP Cluster environment and ensure redundancy, all components of Historical Reporting are deployed at two data centers as shown in the diagram below. One or several supported RDBMS instances are deployed at each data center to host Genesys databases.



At each active data center, multiple instances of Interaction Concentrator collect data regarding contact center activity in the Cluster environment as this data comes in the form of events from SIP Servers that operate in cluster mode. A separate instance of Interaction Concentrator for each data center in the Cluster collects data regarding contact center configuration from Configuration Server Proxy that is located in the respective data center. Another instance of Interaction Concentrator for each data center collects data regarding virtual queue (VQ) DN activity from the VQ SIP Server that is located in the same data center. All data is stored in Interaction Database (IDB), one per Interaction Concentrator server (ICON). Each IDB uses a separate, dedicated database schema hosted at the same data center as the corresponding ICON server. A single Genesys Info Mart server instance extracts data from all available IDBs and stores the data into a single Info Mart database. A single Genesys CX Insights (GCXI) instance provides historical reports for end users.

A second instance of Genesys Info Mart can be deployed in the second data center for Disaster Recovery purposes. The two Genesys Info Mart instances must have the same configuration, including connections to all the IDBs in the Cluster deployment. A standby instance of GCXI can be deployed against the second instance of Genesys Info Mart for Disaster Recovery purposes, but this GCXI instance must *not* be active during normal operations. See Historical Reporting and SIP Business Continuity.}

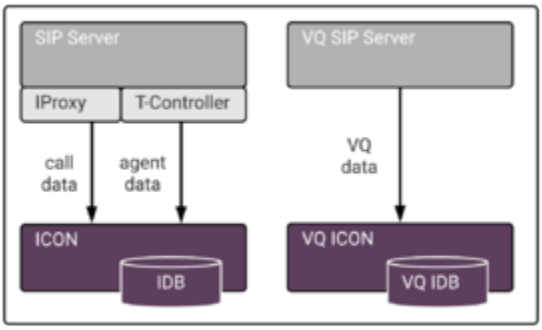
Data Flow Between Components

This section notes the key differences in data flow between components that are specific to a Cluster environment.

Data from SIP Server to Interaction Concentrator

One SIP Server / One Interaction Concentrator / One IDB

In a Cluster environment, one instance of Interaction Concentrator is connected to each SIP Server in the Cluster in order to receive agent and call data, as shown in the diagram below.



ICON recognizes a SIP Server that operates in cluster mode based on the settings in the SIP Server `Application` object.

IProxy and T-Controller are separate interfaces within SIP Server from which ICON collects call data and agent activity data, respectively. By default, a single ICON handles both data types, as shown in the diagram above, and then writes all data, along with respective user data, to a single IDB. Depending on the call flow, SIP Server in one SIP Cluster node may process call data while SIP Server in another SIP Cluster node maintains agent activity data for the same call. In this case, two different ICONs that belong to respective SIP Cluster nodes independently write data for a single call into two different IDBs, one storing call data and the other, agent activity data. The VQ SIP Server should also have a dedicated instance of Interaction Concentrator (ICON server and IDB) associated with it. This ICON processes data in the same way as in a non-Cluster environment. See Configure and install the Interaction Concentrator (ICON) application on the Enabling Historical Reporting topic for the necessary configuration procedure.

Because an ICON that is connected to a SIP Server in a Cluster uses different connection parameters and expects different data than in a non-Cluster environment, keep in mind the following requirements:

- An ICON that is connected to a SIP Server that is a part of a Cluster must not connect to any T-Server or non-Cluster SIP Server.
- Each ICON application must populate its own IDB. In other words, consider each ICON-IDB pair a unit (Interaction Concentrator instance).
- No other ICON should write data to the IDB that is used by the ICON connected to a SIP Server that operates in Cluster mode.

Data Flow from IDBs to Genesys Info Mart

Genesys Info Mart retrieves data from IDBs in a Cluster environment in the same way that it retrieves data in a non-Cluster environment. Using the same extraction logic as applies to any high availability (HA) environment, Genesys Info Mart selects the IDB with better data quality out of the two IDBs in each HA pair. Genesys Info Mart associates call, agent activity, and user data with a given call in the case these types of data come from different IDBs that store events from different SIP Cluster nodes. A single Info Mart database stores the data extracted from all IDBs in the Cluster deployment.

Data Flow from Info Mart Database to GCXI Reports

The data for GCXI reports is available from the Info Mart database through the Reporting and Analytics Aggregates (RAA) module, in the same way as in a non-Cluster environment.

Historical Reporting Deployment Considerations

The SIP Cluster deployment must be set up and provisioned according to Deploying SIP Cluster prior to deploying Historical Reporting components. The Historical Reporting components themselves must be deployed per recommendations in this article and following the procedure described in Enabling Historical Reporting.

Scalability

Historical Reporting architecture scales up with the rest of SIP Cluster deployment:

- One Interaction Concentrator instance (or HA pair) must be deployed per SIP Cluster node.
- One Interaction Concentrator instance (or HA pair) must be deployed per VQ SIP Server (or HA pair of VQ SIP Servers).
- One Interaction Concentrator instance (or HA pair) must be deployed per data center, to collect configuration details from Configuration Server Proxy.
- One Interaction Concentrator instance (or HA pair) must be deployed per OCS instance, to store Outbound details.
- One Genesys Info Mart instance must be deployed to extract data from all Interaction Databases (IDBs) in the SIP Cluster deployment.

Configuration Prerequisites

Two SIP Server Application objects must be created per SIP Server running in Cluster mode, as described in Configuring SIP Servers for Historical Reporting, to enable ICON connections to the two different SIP Server ports. See also User Data, below.

Data Differences in IDB

Data processing logic that is specific to a Cluster environment accommodates independent processing of call data and agent activity data coming from different SIP Servers for the same voice interaction. This processing results in a number of noteworthy differences in the types of data that Interaction Concentrator produces and stores in IDB, as listed in “Table Fields That Differ in a Cluster Environment,” below. Most importantly, the G_PARTY table does not contain the agent ID value. Additionally, the value for the PartyID has a different format and is calculated differently than in a non-Cluster environment.

Table Fields That Differ in a Cluster Environment

The AgentID field in the following table contains the value 0:

- G_PARTY

The EndpointID field in the following tables contains a NULL or 0 value:

- GC_IVRPORT
- GS_AGENT_STAT
- GS_AGENT_STAT_WM
- GX_SESSION_ENDPOINT
- G_AGENT_STATE_HISTORY
- G_AGENT_STATE_RC
- G_DND_HISTORY
- G_PARTY
- G_PARTY_HISTORY
- G_SECURE_USERDATA_HISTORY
- G_USERDATA_HISTORY

The GSYS_EXT_VCH1 field in the following tables contains the DN name:

- GS_AGENT_STAT
- GS_AGENT_STAT_WM
- GX_SESSION_ENDPOINT
- G_AGENT_STATE_HISTORY
- G_AGENT_STATE_RC
- G_DND_HISTORY
- G_LOGIN_SESSION
- G_PARTY_HISTORY

The DestEndPointID field in the following table contains a NULL or 0 value:

- G_ROUTE_RESULT

The DestEndPointType field in the following table contains the value 1:

- G_ROUTE_RESULT

The EndPointType field in the following tables contains a value of 1:

- G_PARTY

- GX_SESSION_ENDPOINT

The LoginID field in the following tables is NULL:

- G_AGENT_STATE_HISTORY
- G_LOGIN_SESSION

The PlaceID field in the following tables is NULL:

- GX_SESSION_ENDPOINT
- G_AGENT_STATE_HISTORY
- G_LOGIN_SESSION

Data Differences in Info Mart

Most of the Info Mart tables and views are populated similarly in Cluster and non-Cluster environments. The Info Mart data differences in a Cluster environment result from differences in IDB data. Some dimensions (such as RESOURCE_) are populated during transformation of the voice or agent-related data while some others might be not populated at all (such as PLACE). As a result, only a subset of DN-based metrics is available in a Cluster environment.

The following subsections provide more details about the populated data. For illustration purposes, it is assumed that the T-Servers in the environment are a mix of SIP Servers that operate in Cluster mode and either non-Cluster SIP Servers or traditional T-Servers.

GIDB Data

Global Interaction Database (GIDB) is the part of the Info Mart database that is designed to keep all records that are extracted from various IDBs. As such, the GIDB data that is available in the Cluster environment reflects the changes in the corresponding IDB data. See Data Differences in IDB, above.

Resources and Resource Groups

The resource-related tables and views are populated as described below.

Table RESOURCE_

Population of the RESOURCE_ table differs with regard to SIP Cluster resources (endpoint DNs). When Genesys Info Mart encounters SIP Cluster resources during voice or agent-state transformation, the transformation job creates resource records "on the fly."

In the RESOURCE_ table:

- For IVR applications that are not represented in the Configuration Layer (that is, whose configuration type equals 0), records are created based on the voice and agent-related data. Genesys Info Mart continues to identify them by the combination of the RESOURCE_NAME and TENANT_KEY.
- For the resources controlled by SIP Servers operating in Cluster mode, records are created based on the

voice and agent-related data. Genesys Info Mart identifies them by the combination of RESOURCE_NAME and SWITCH_DBID.

When it creates the RESOURCE_ record, Genesys Info Mart generates an internal ID to populate the RESOURCE_CFG_DBID field. The RESOURCE_CFG_TYPE_ID field is then set to 0 (zero).

For any other resources (such as agents, queues, Routing Points, etc.), records are created based on the configuration data. Genesys Info Mart continues to identify them by the combination of configuration type and DBID (stored in the RESOURCE_CFG_TYPE_ID and RESOURCE_CFG_DBID in the RESOURCE_ table).

Table GROUP_TO_CAMPAIGN_FACT

The GROUP_TO_CAMPAIGN_FACT table is populated based on the data from the Configuration detail IDB. In an environment where Genesys Outbound Contact is deployed, this table contains records for campaign groups, which are configured as agent groups in the Configuration Layer.

View GROUP

The data in the GROUP view reflects the groups of DNs and agents as configured in the Configuration Layer.

Places and Place Groups

With Place objects not being configured in a Cluster environment, the Place-related table and view are populated as described below.

Table PLACE_GROUP_FACT

The PLACE_GROUP_FACT table contains records only if Place objects are configured in the Configuration Layer. With the absence of Place objects in a Cluster environment, this table might not contain any data.

View PLACE

The data in the PLACE view contains records only if Place objects are configured in the Configuration Layer. With the absence of Place objects in a Cluster environment, this view might not contain any data.

User Data

For the purposes of Historical Reporting, user data configuration and processing do not differ in a Cluster environment. However, to enable Genesys Info Mart to extract user data from both I-Proxy and T-Controller channels, ICON application requires a connection configured to a special (dummy) SIP Server application that specifies connection parameters of the T-Controller.

Historical Reporting Operational Considerations

Operational Considerations for ICON in a Cluster Environment

Filtering Event Subscription

To ensure that network traffic is as efficient as possible, ICON in a Cluster environment requests that only a subset of the possible TEvents be sent by SIP Server.

User Data Subscriptions

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. To lighten network load, by default ICON collects only a certain set of user data types.

To add other types, configure the **ccon_adata_spec.xml** file, as described in Enabling Storage of User Data. For a detailed discussion of attached data, see Processing Attached Data in the *Interaction Concentrator User's Guide*.

Operational Optimization

To optimize operations and prevent data loss, additional functionality is implemented in Interaction Concentrator. Improvements in the following areas ensure that ICON observes stricter startup conditions and sustains short network disconnects:

PQ File

To prevent data loss at the pq file level, ICON does not start and does not attempt to connect to SIP Server if the pq file is not created or opened correctly on startup. In case of a problem, a Standard-level log event (ID# 09-25024) notes the error.

Event Backlog

To prevent data loss during a temporary disconnection from a client such as ICON, SIP Server stores all events in its cache and automatically sends them to the client when the connection is restored.

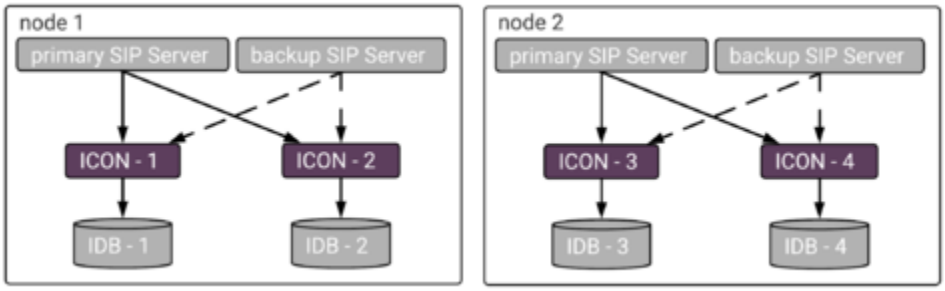
High Availability

As in a non-Cluster environment, Interaction Concentrator supports high availability (HA) deployments of SIP Server. HA for the SIP Server that operates in Cluster mode functions in the same way as for the non-Cluster SIP Server. A number of HA options are available for SIP Server, as documented in the SIP Server High-Availability Deployment Guide.

Redundant Deployment of Interaction Concentrator

Interaction Concentrator has no specific features for HA deployment in a Cluster environment. As in a non-Cluster environment, a redundant pair of ICONs receives interaction-related events from the same SIP Server. Both ICONs in the HA pair operate in parallel as stand-alone servers; they process incoming data independently and store data in two independent IDBs. The downstream reporting applications are responsible for managing extraction of reliable data.

As a general recommendation, deploy one HA pair of Interaction Concentrator instances per SIP Server (or HA pair of SIP Servers). As shown in the diagram, each Interaction Database instance should be located at the same site as the ICON Server writing to it.



For a complete description of Interaction Concentrator HA functionality, see The Interaction Concentrator HA Model in the *Interaction Concentrator User's Guide*.

Disaster Recovery

In a Cluster environment, Interaction Concentrator is set up locally with each SIP Server in a cluster node. In a disaster recovery scenario, calls move to SIP Servers in the Cluster nodes within the still-active data centers and the ICON instances located in those nodes store data in the associated IDBs as usual. You can replicate IDBs for additional data redundancy, if required.

Operational Considerations for Genesys Info Mart in a Cluster Environment

The differences in data processing for Genesys Info Mart operating in a SIP Cluster environment arise from the need to match up agent and interaction information from separate ICON providers for I-Proxy and T-Controller data, respectively.

Matching Agents with Interaction Information

In all deployments, agent information and interaction information for Genesys Info Mart come from different ICON providers (gls and gcc). In a non-Cluster deployment, the gcc provider reports the AGENTID in the party record. In the SIP Cluster solution, there is no AGENTID in the party record because ICON receives call data separately from agent activity data. Additional logic enables the transformation job to use endpoint and timestamp information to identify the AGENTID from the

G_LOGIN_SESSION table.

> **Important**
>
> When it is deployed in a mixed environment, Genesys Info Mart extracts both Cluster-related and non-Cluster-related data from different IDBs and uses the appropriate logic for processing available data.

High Availability

There are no special requirements for Genesys Info Mart to support HA of Interaction Concentrator and upstream data sources in SIP Cluster deployments. Similarly, there are no special features of Genesys Info Mart HA support that apply for SIP Cluster.

Disaster Recovery

Genesys Info Mart does not provide HA of Genesys Info Mart itself. However, you can deploy a second instance of the Genesys Info Mart Server, along with a second instance of the Info Mart database, for Disaster Recovery purposes. See Historical Reporting and SIP Business Continuity.

Enabling Historical Reporting

The following task summaries provide an overview of the steps that are required to deploy the products that provide Historical Reporting in the SIP Cluster solution. In general, there are no special steps; only a few configuration settings are specific to the Cluster deployment.

Deploying Interaction Concentrator and Genesys Info Mart

Task Summary

1. **Plan the deployment.**
   For the Interaction Concentrator and Genesys Info Mart architectures that are supported for the SIP Cluster solution, see Historical Reporting Architecture.
2. **Configure Host configuration objects.**
   Configure a Host configuration object for each computer on which the DB Server, ICON server, and Genesys Info Mart server applications will reside.
3. **Assign a DB Server to enable IDB storage.**
   Unless you plan to use a DB Server that also serves another application, configure and install a DB Server for IDB storage purposes. For performance reasons, Genesys recommends that you set up the DB Server on the same host as the RDBMS server.
   For guidelines regarding the number of DB Servers in an environment with multiple IDB instances, refer to Deploying DB Server in the *Interaction Concentrator Deployment Guide*.
4. **Configure and install the Interaction Concentrator (ICON) application.**
   Settings for certain options in the **[callconcentrator]** and **[filter-data]** sections affect Genesys Info Mart and, in deployments that include it, Genesys-provided aggregation. For more information, see Preparing the ICON Application in the *Genesys Info Mart Deployment Guide.*

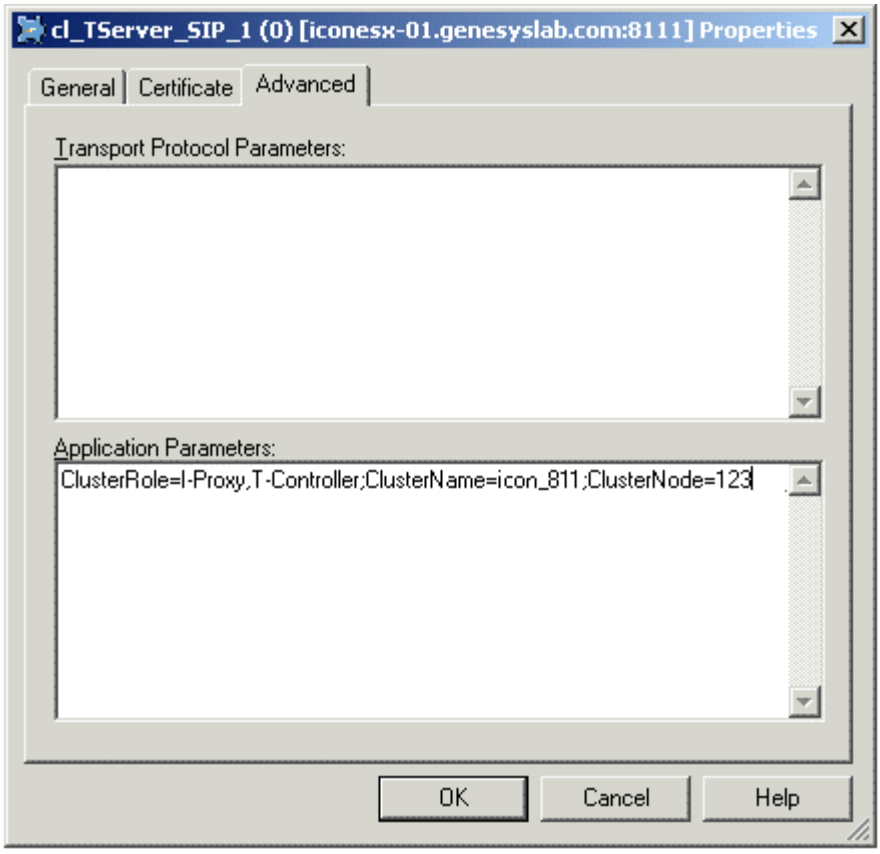
   1. The following options in the **[callconcentrator]** section on the ICON `Application` object, with the required settings as noted below, control how Interaction Concentrator stores data in a Cluster environment:
      - cluster-iproxy-udata = `all` (or `conf` if only some user data keys are to be saved)
      - use-server-partyuuid = 1
      - ph-use-epn = 1
   2. Genesys Info Mart requires the following settings for options in the **[callconcentrator]** section on the ICON `Application` object:
      - use-dss-monitor = `true`
      - calls-in-the-past = `true`
      - om-force-adata = `true`
      - gls-active-reason-codes = `true`
      - partition-type = 2

- role = `cfg` (for the Configuration details ICON)
- role = `gcc,gud,gls` (for the Voice details ICONs and VQ ICONs)
- role = `gos` (for the Outbound Contact details ICONs)

3. Deploying Interaction Concentrator in a Cluster environment requires some additional configuration on the **Connections** tab. Configure the **Connections** tab according to the instructions in the *Interaction Concentrator Deployment Guide*, but making the following adjustments specific to a SIP Cluster environment:
   1. Configure a connection to Configuration Server Proxy instead of configuring a connection to Configuration Server.
   2. Add the SIP Server to which this instance of Interaction Concentrator should connect.
      If the instance of ICON you are configuring is associated with the VQ SIP Server, no further configuration is necessary. If you are connecting to a SIP Server running in a Cluster mode and handling call and agent data, perform the following steps:
      1. Add a second, dummy SIP Server `Application` object to the ICON `Application` **Connections** tab. This Application should have the same configuration as the actual SIP Server `Application` object except as noted in Configuring SIP Servers for Historical Reporting.
      2. Select a SIP Server `Application` from the **Connections** list and click **Edit**.
      3. If configuring connection to the actual SIP Server application, specify connection parameters of the IPport. If configuring connection to the dummy SIP Server application, specify connection parameters of the TCport.
      4. On the **Advanced** tab of the **Connection Info** dialog box, configure the following parameters (shown in the graphic below):
         - **ClusterRole**
           Valid Values:
           - `I-Proxy` - Interaction Concentrator processes interaction data from the SIP Server that operates in Cluster mode. Use this value in the actual SIP Server `Application` object.
           - `T-Controller` - Interaction Concentrator processes agent activity data from the SIP Server that operates in Cluster mode. Use this value in the dummy SIP Server `Application` object.

           Default Value: None

           Note: For details on the supported architecture, see Historical Reporting Architecture.
         - **ClusterName** - The name of the Interaction Concentrator `Application`.
         - **ClusterNode** - The DBID of the Interaction Concentrator as stored in the Configuration Layer.



After completing **Connections** tab configuration for ICON connections to both actual and dummy SIP Server Applications, continue to Step 5 (below).

- **Create a new database for each IDB instance that you intend to deploy for ICON data storage.**
  After creating the IDBs, initialize each IDB instance, as described in Deploying IDB in the *Interaction Concentrator Deployment Guide*.
- **For each ICON, configure a database access point (DAP) Application object that specifies IDB connection parameters.**
  Follow the procedure for Deploying DAP in the *Interaction Concentrator Deployment Guide*.
  Ensure that the **role** option values that you specify for the DAP are consistent with the **role** option values specified for the ICON instance that it serves.
- **Modify the attached-data specification file to enable capture and storage of the attached data that you require.**
  See Enabling Storage of User Data, below.
- **Ensure that all required data sources have been enabled, to identify them to Genesys Info Mart as available.**
- **Prepare the Genesys Info Mart database and views.**
  Create and configure database schemas to process and store detailed reporting data. You can find the database scripts required for your RDBMS in the following folder on the Genesys Info Mart product CD:
  `genesys_info_mart\db_scripts\<operating_system>\sql_scripts\<RDBMS_type>`

  For full information, see the sections Preparing the Info Mart Database, Optimizing Database Performance: Database Links, and Optimizing Database Performance: Database Tuning in the *Genesys Info Mart Deployment Guide.*

After you have installed Genesys Info Mart, create tenant views if necessary (see step 18, below).

- **Configure the DAPs that Genesys Info Mart uses to access source and target databases.**
  You need DAPs for:
  - Genesys Info Mart Server to access the IDBs (extraction DAPs).
  - Genesys Info Mart Server to access the Info Mart database (Info Mart DAP).
  - The Administration Console to access the Info Mart database (Administration Console DAP), if you plan to use the Genesys Info Mart Administration Console to monitor ETL jobs. No DAP is required for GIM Manager.

  For the extraction DAPs, you can reuse an existing ICON DAP; to do so, add the gim-etl section with the configuration options that Genesys Info Mart requires in a DAP Application object. For more information about creating new DAPs or reusing existing ones, see Configuring Required DAPs in the *Genesys Info Mart Deployment Guide.*
- **Configure the Genesys Info Mart Server application.**
  The required configuration settings depend directly on the Genesys Info Mart features that you want to implement and on your choice of an end-user reporting tool, such as Genesys CX Insights (GCXI). For full information, see Configuring the Genesys Info Mart application in the *Genesys Info Mart Deployment Guide.* There are no special configuration requirements for the SIP Cluster solution. However, as with any deployment with distributed data centers, if you do decide to configure an optional overt connection to Configuration Server, configure the connection to Configuration Server Proxy instead of to Configuration Server. If your deployment will include Genesys-provided aggregation, be aware that settings on the Genesys Info Mart application affect aggregation. For full information about how to configure Genesys Info Mart for aggregation, see the *Reporting and Analytics Aggregates Deployment Guide.*
- **Configure Switch and DN objects as required for use by ICON or Genesys Info Mart.**
  Options in the gts section (for ICON) and the gim-etl section (for Genesys Info Mart) either do not apply in the SIP Cluster solution or else are required to have the same values as would be set in a non-SIP Cluster deployment, as described in Configuring Supporting Objects in the *Genesys Info Mart Deployment Guide.* For example, in the case of options in the gts section on the Switch object, only the options whose area of functionality is identified as agent-related (Agent State and Login Session, Agent Metrics) apply in the SIP Cluster solution; the gls-associations-rule option, if configured, must be set to `0`, and the gls-use-ts-id option, if configured, must be set to 1.
- **Prepare the Genesys Info Mart Server host.**
  Genesys Info Mart requires Java 1.7 JDK or Server JRE, at a minimum, but version 1.8 is recommended. For information about installing the required JDK and JDBC driver, updating system information, and setting up for the optional use of Transport Layer Security (TLS) for connections to Configuration Server Proxy and Message Server, see Preparing the Genesys Info Mart Server host in the *Genesys Info Mart Deployment Guide.*
- **Install the Genesys Info Mart application on its host.**
- **Install the GUI you plan to use to manage and monitor Genesys Info Mart jobs.**
  Depending on your configuration GUI, you can use either:
  - (Recommended) Genesys Info Mart Manager (GIM Manager)—A Genesys Administrator Extension (GAX) plug-in. For information about installing this component, see Installing Genesys Info Mart Manager.
  - The Genesys Info Mart Administration Console—An extension to Genesys Configuration Manager that uses the Wizard Framework. You must install the Administration Console GUI on the same host on which Configuration Manager is installed. The Administration Console host must use the Microsoft Windows operating system.
- **Start Genesys Info Mart.**

- **Verify the deployment.**

  1. Review Genesys Info Mart logs to verify that the deployment is complete and configuration is correct.
  2. In your management GUI (GIM Manager or the Administration Console), review the status of **Job_InitializeGIM** to verify successful initialization of the database and successful update of the IDBs.

- **Create tenant-specific, read-only views on the Info Mart database.**
  Tenant views are strictly required only for multi-tenant deployments; however, Genesys recommends configuring read-only views for single-tenant deployments as well. For more information, see Creating Read-Only Tenant Views in the *Genesys Info Mart Deployment Guide.*
- **Access the management GUI, to continue managing Genesys Info Mart jobs.**

  - For information about accessing GIM Manager, see Using Genesys Info Mart Manager in the *Genesys Info Mart Operations Guide.*
  - For information about accessing the Administration Console, see the chapter about post-installation activities in the *Genesys Info Mart 8.1 Deployment Guide.*

  For more information about how to manage jobs, see Managing and Scheduling Jobs in the *Genesys Info Mart Operations Guide.*
- **(Optional) Enable aggregation (see Enabling Aggregation).**
  Note: This is required for GCXI.

Enabling Storage of User Data

Task Summary

1. **Configure the ICON application to store user data.**
   Ensure that the following option values are set:
   - adata-extensions-history = none
   - adata-reasons-history = none
   - adata-userdata-history = none
2. **Specify the user data that ICON will store in IDB.**
   1. Identify the key-value pairs (KVPs) from various applications that Genesys Info Mart requires for data processing.
   2. For call-based attached data, modify the ICON attached data specification file to capture the KVPs that you require and to control in which IDB table(s) ICON will store the data. If necessary, modify the ICON adata-spec-name configuration option to match the file name that you use.
      Genesys Info Mart provides a sample specification file, **ccon_adata_spec_GIM_example.xml**, which includes the KVPs for which storage is predefined in Genesys Info Mart.
   3. For UserEvent-based user data, set ICON configuration options in the **[custom-states]** section, as required. For more information, see the Interaction Concentrator **[custom-states]** options.
3. **Plan the Info Mart tables in which you want to store custom user data.**

Identify the Info Mart fact or dimension tables in which you want custom user data to be stored, and map the user-data KVPs to the Info Mart tables and columns that you have identified. For more information, see Storing User Data and User Data Mapping in the *Genesys Info Mart Deployment Guide.*

4. **Modify the Info Mart database schema to store the custom user data.**
   For information about creating the custom tables and columns and mapping the KVPs to them, see Preparing Custom User-Data Storage in the *Genesys Info Mart Deployment Guide.* You execute the customized script when you create the rest of the Info Mart database schema, or when you complete the deployment after installing Genesys Info Mart.
5. **(Optional) Enable storage of user data for interactions that are in mediation.**
   In the applicable Virtual Queue DN objects, configure the link-msf-userdata configuration option in the gim-etl section (link-msf-userdata=true).
6. **If necessary, turn off Genesys Info Mart filtering of user data in IDB.**
   By default, Genesys Info Mart filters the call-based attached data in IDB to extract KVPs that were sent in the UserData event attribute only. If you have configured the attached data specification file for ICON to store custom KVPs from the Reasons or Extensions attributes as well, turn off Genesys Info Mart filtering by setting the Genesys Info Mart **filterUserData** startup parameter to false. For information about changing this startup parameter, see Modifying JVM Startup Parameters in the *Genesys Info Mart Deployment Guide.*
7. **(Optional) Streamline Genesys Info Mart processing of user data.**
   If your historical reporting involves the use of large quantities of user data, consider increasing the value of the ud-io-parallelism option in the gim-transformation section of the Genesys Info Mart Application object.

Enabling Aggregation

Task Summary

1. **Install the aggregation engine software—Reporting and Analytics Aggregates (RAA) package.**
   RAA is not included with the GCXI installation package, as it was with the legacy GI2 installation package. For full information about how to install GCXI, including requirements to the machine that hosts Microstrategy software, see the *Genesys CX Insights Deployment Guide*.
   - For full information about how to install the RAA package, see the *Reporting and Analytics Aggregates Deployment Guide*.
2. **Verify that ICON, Genesys Info Mart, and other objects have been configured as required for aggregation.**
   For more information, see the *Reporting and Analytics Aggregates Deployment Guide*.
3. **Configure GCXI.**
   For more information, see the *Genesys CX Insights Deployment Guide*.
4. **(Optional) Configure custom calendars.**
   For more information, see Creating Custom Calendars in the *Genesys Info Mart Deployment Guide.*
5. **Start the aggregation engine.**
   If you have configured the Genesys Info Mart scheduler to control the aggregation process (run-aggregates=true), the aggregation job will start automatically at the scheduled time (as specified by aggregate-schedule).
   - For full information about how to start the aggregation engine, see How Do I Control Aggregation at

Runtime? in the *Reporting and Analytics Aggregates Deployment Guide.*

- For more information about how to schedule and manage the aggregation job in Genesys Info Mart, see Managing and Scheduling Jobs in the *Genesys Info Mart Operations Guide.*

Historical Reporting and SIP Business Continuity

Historical Reporting supports SIP Business Continuity in the SIP Cluster Solution through deployment of an active Genesys Info Mart instance, along with an operational Info Mart database, at each of the two data centers. Genesys recommends the active-active architecture for Genesys Info Mart because it minimizes data loss at Disaster Recovery, requires smaller network bandwidth, and requires no replication software, which is often expensive in deployment and maintenance.

For more information about Genesys Info Mart support for Business Continuity environment in general, refer to the Genesys Info Mart 8.1 Business Continuity Deployment Guide. For important information that is not covered in this *Solution Guide,* see, in particular, the sections on Potential Data Loss and Disaster Recovery Procedure.

Architecture

Two Genesys Info Mart Server instances are active at two data centers; database replication is not required between the two instances. In other words, the two Genesys Info Mart Servers operate in parallel as stand-alone servers at their respective data centers. Both Genesys Info Mart Servers access all the Interaction Databases (IDBs) at both data centers and, for each HA pair, extract data from the IDB that has the best quality data from a particular data source. Each Genesys Info Mart Server stores extracted data at its own Info Mart database independently. Both Info Mart databases, therefore, contain data that is nearly identical and that reflects activity of the entire contact center.

To achieve as close similarity of data as possible, both Genesys Info Mart Servers must have identical configuration. Even then, the data sets in two databases would differ because the data is processed independently and, consequently, the numbers of data keys would be unique in each Info Mart database.

The data from Genesys Info Mart at data center 2 is not used by downstream Reporting applications unless data center 1 fails. An active instance of Genesys CX Insights (GCXI) at data center 1 retrieves data from the Info Mart database at data center 1 to provide historical reports for all users. A standby GCXI instance at data center 2 can be brought into service in the event that data center 1 fails.

Disaster Recovery Procedure

In the event that data center 1 fails, the disaster recovery procedure must be started. As part of this procedure:

- Genesys Info Mart at data center 2 is configured to ignore data sources at the failed data center 1.
- GCXI instance is brought into service at data center 2.

When the failed data center, or its replacement, is back in service, another active Genesys Info Mart

instance can be added to the deployment again.

SIP Cluster-specific Functionality

This section provides descriptions of new features introduced in SIP Server 8.1.103.12 and later.

- Release 8.1.103.58: Troubleshooting T-Controllers communication issues
- Release 8.1.103.28: Disabling recording and monitoring of outbound calls
- Release 8.1.103.26: Enabling call recording on the agent side
- Release 8.1.102.95: Call supervision during IVR phase

Disabling recording and monitoring of outbound calls

Starting with release 8.1.103.28, per GDPR compliance, SIP Server can disable recording and monitoring of outbound calls in SIP Cluster deployments. This feature is only supported for TMakeCall requests made through a Routing Point and then routed to an external number.

Sample Call Flow

1. An agent dials to an external number from a Workspace Web Edition (WWE) desktop.
2. WWE translates the destination and submits a TMakeCall request.
3. SIP Server receives the TMakeCall request and queues the call on the Routing Point.
4. The strategy on the Routing Point issues a TRouteCall request containing **make-call-cpd-required** = `true` and **UseDialPlan** = `agentid` in AttributeExtensions.
5. SIP Server processes the call destination and routes the call to the Trunk Group DN. The agent is connected with the Trunk Group DN. This is similar to the engaging call in Active Switching Matrix (ASM) mode.
6. SIP Server issues an internal TApplyTreatment request for playing standard music to the agent.
7. SIP Server issues an internal TMakePredictiveCall request with the Call Progress Detection (CPD) extensions and the IVR profile to be selected.
8. Once the agent is connected, SIP Server instructs a media server to perform the CPD. This is similar to the outbound call in ASM mode.
9. After the CPD is completed, SIP Server issues TApplyTreatment to play the prepared IVR profile. This profile contains the VXML script that plays the recording or monitoring opt-out prompt and collects the user input on it.
10. SIP Server merges both call parties by issuing TMergeCall.
11. On the completion of the merge call process, the agent and called party are connected.

Feature Configuration

1. Configure a Trunk Group DN for CPD functionality in the SIP Cluster switch. In this Trunk Group DN, add the following configuration options:
   - **ivr-profile-name**: the IVR profile that points to the VXML page responsible for a recording and monitoring opt out prompt.
   - **beep**: whether to apply a beep tone before connecting the agent from the engaging call with the called party from the outbound call.
   - **predictive-call-timeout**: the value to be included as an AttributeTimeout value in the TMakePredictiveCall request.

- **cpd-extensions**: a list of CPD-related AttributeExtensions that are included in TMakePredictiveCall.

2. Set the **make-call-cpd-dn** option to the Trunk Group DN (created in Step 1) on the Routing Point DN, where TMakeCall requests submitted by agents are queued.
3. Set the **make-call-cpd-required** key extension to `true` in TRouteCall.
4. If required, on the Trunk Group DN, set **AnsMachine**, **FaxDest**, and/or **SilenceDest** keys in the Extensions attribute of TMakePredictiveCall requests. For this feature, a value of `connect` (in addition to `drop`) is supported. These keys override respective Application-level configuration options: **am-detected, fax-detected, silence-detected**.

AttributeExtensions

Key: **make-call-cpd-required**
Type: String
Valid Values: `true, false`
Request: TRouteCall

Specifies whether SIP Server applies CPD functionality to the specified TRouteCall destination.

- If set to `true`, SIP Server applies CPD functionality to the specified TRouteCall destination by converting this request to the Active Switching Matrix (ASM) mode call flow.
  Note: Genesys recommends setting the **UseDialPlan** extension to `agentid` when **make-call-cpd-required** is set to `true`. SIP Server connects an agent with the configured Trunk Group DN and for the specified destination CPD is done through TMakePredictiveCall. Thus, the dial plan is not required and only an agent ID provided by SIP Feature Server is added to the response.
- If set to `false`, SIP Server performs the routing request and no CPD functionality is applied.

If this extension is not specified, then CPD functionality is not applied.

Configuration Options

make-call-cpd-dn

Setting: Routing Point DN > [TServer] section
Default Value: An empty string
Valid Values: Any Trunk Group DN
Changes Take Effect: For the next call

When this option is set to a valid Trunk Group DN, SIP Server invokes CPD functionality for the call routed to an external number. The call is queued on a Routing Point when a TMakeCall request is made by an agent.

This option can be configured on the Routing Point DN in the SIP Cluster switch. Or, this option can be specified in the SIP Cluster DN (the DN with **service-type** set to `sip-cluster-nodes`) to apply to all Routing Point DNs under a SIP Cluster switch. The Routing Point DN setting takes precedence over the SIP Cluster DN setting.

ivr-profile-name

Setting: Trunk Group DN > [TServer] section
Default Value: An empty string
Valid Values: Any valid IVR profile
Changes Take Effect: For the next call

Specifies the name of the IVR profile that is added in the Request-URI sent to the Media Server.

Sample URI format: sip:msml@<RM:FQDN>;media-service=cpd;gvp-tenant-id=<ivr-profile-name>

beep

Setting: Trunk Group DN > [TServer] section
Default Value: `on`
Valid Values: `on, off`
Changes Take Effect: For the next call

When set to `on`, SIP Server applies a beep tone for the specified duration to the agent before processing the TMergeCall request. When set to `off`, SIP Server merges both the engaging and outbound call without playing a beep tone.

predictive-call-timeout

Setting: Trunk Group DN > [TServer] section
Default Value: `20`
Valid Values: `0-1800`
Changes Take Effect: For the next call

Specifies, in seconds, the value to be included as an AttributeTimeout value in the TMakePredictiveCall request. If this timeout expires before the call is answered, or if SIP Server receives a BYE message from the Media Server, SIP Server terminates the call.

This AttributeTimeout value is applied only when the **predictive-timerb-enabled** option is set to `false` in the Trunk Group DN. When **predictive-timerb-enabled** is set to `true`, SIP Server uses the 32-second timer and ignores the timeout specified in this option.

cpd-extensions

Setting: Trunk Group DN > [TServer] section
Default Value: An empty string
Valid Values: A comma-separated key-value pairs of TMakePredictiveCall without spaces
Changes Take Effect: For the next call

Specifies CPD-related AttributeExtensions to be included in TMakePredictiveCall. SIP Server applies default values to the non-configured extensions. For example:
`cpd-record=on,call_answer_type_recognition=positive_am_detection,cpd-on-connect=off,call_timeguard_timeout=2000,AnsMachine=connect,FaxDest=drop,SilenceDest=drop`

CPD AttributeExtensions

Key: **cpd-record**
Default Value: `off`
Valid Values: `on, off`
Request: TMakePredictiveCall
Enables or disables the recording of the call progress detection phase of the call.

Key: **call_answer_type_recognition**
Type: String
Default Value: `positive_am_detection`
Valid Values: `no_progress_detection, no_am_detection, positive_am_detection, full_positive_am_detection, accurate_am_detection, telephony_preset`
Request: TMakePredictiveCall
Specifies answer, answering machine, and fax detection settings when dialing using SIP Server.

Key: **cpd-on-connect**
Type: String
Default Value: `off`
Valid Values: `on, off`
Request: TMakePredictiveCall
Specifies when call progress analysis is started.

Key: **call_timeguard_timeout**
Type: String
Default Value: `3000`
Valid Values: Time interval in msec
Request: TMakePredictiveCall
Enables setting a timeout for post-connect call progress detection. The call is transferred to a queue when the timeout expires, regardless of the call result or the completion of call progress detection.

Key: **AnsMachine**
Type: String
Valid Values: `connect, drop`
Request: TMakePredictiveCall
Specifies whether SIP Server connects or drops a call if the CPD result shows that the predictive call reached an answering machine.

Key: **FaxDest**
Type: String
Valid Values: `connect, drop`
Request: TMakePredictiveCall
Specifies whether SIP Server connects or drops a call if the CPD result shows that the predictive call reached a fax machine.

Key: **SilenceDest**
Type: String
Valid Values: `connect, drop`
Request: TMakePredictiveCall
Specifies whether SIP Server connects or drops a call if the CPD result shows that silence is detected.

Feature Limitations

- This feature supports only direct outbound calls made by an agent to an external destination through TMakeCall requests.
- Outbound calls to an external destination through T-Library requests—TSingleStepConference, TInitiateConference, TSingleStepTransfer, TInitiateTransfer—are not supported.
- If a Trunk Group DN is configured with **call_answer_type_recognition**=`no_progress_detection`, CPD analysis is not done, but SIP Server still generates TApplyTreatment to play the prepared IVR profile. This is considered as misconfiguration.

Enabling call recording on the agent side

Starting with version 8.1.103.26, SIP Server can enable call recording on the agent side in SIP Cluster deployments.

To enable this feature:

1. Configure the **record** option to `true` in the Person object associated with the agent.
2. Configure the **request-person-options** option to `true` in the VoIP Service DN containing the **service-type** option set to `sip-cluster-nodes`.

In standalone mode, Agent Login objects are used to configure agent-based recording (record=true). In SIP Cluster deployments, Agent Login objects are not used.

Configuration Options

record

Setting: Annex tab, TServer section, in the Person object associated with the agent
Default Value: `false`
Valid Values: `true, false`
Changes Take Effect: When the next call is established on the DN

When set to `true` and when a call is established on the DN on which this person has logged in, recording starts. SIP Feature Server reads this option and adds the option setting to the XS Dialplan Response. SIP Server reads the Dialplan Response and processes the call recording.

request-person-options

Setting: Annex tab, TServer section, in the VoIP Service DN containing the **service-type** option set to `sip-cluster-nodes`
Default Value: `true`
Valid Values: `true, false`
Changes Take Effect: When the next call is established

When set to `true`, SIP Server adds the `request-person-options` tag to the XS Dialplan Requests that are sent to SIP Feature Server.

Troubleshooting T-Controllers communication issues

Introduced in SIP Server version 8.1.103.58. To troubleshoot T-Controllers (TC) communication issues, the following statistics, based on periodic T-Library polls between T-Controllers, are now available:

- T-Lib communication round-trip time (RTT): Time elapsed between a T-Lib request and a response event
- T-Lib request processing latency (TXT): Time elapsed between a T-Lib poll request sent to a poll initiator and received on a poll destination
- T-Lib response processing latency (RXT): Time elapsed between a T-Lib poll response event sent to a poll destination and received on a poll initiator
- The most recent timestamp of a poll request response received for (LAST_TS): The timestamp of the most recent answered poll request
- Current outgoing T-Lib connection output queue length (TX_QUEUE_LEN): Low-level statistic on how much data (in bytes) queued to be sent from this TC to another TC through client connection (this TC to another TC)
- Current incoming T-Lib connection output queue length (RX_QUEUE_LEN): Low-level statistic on how much data (in bytes) queued to be sent from this TC to another TC through server connection (another TC to this TC)

You can configure the interval between poll cycles and TX_QUEUE_LEN/RX_QUEUE_LEN statistics using the tc-latency-poll-interval configuration option.

Call supervision during IVR phase

Introduced in SIP Server version 8.1.102.95, calls in a SIP Cluster deployment can be observed while in the IVR phase where different treatments can be applied.

An IVR supervision session can be initiated by a T-Library client, such as Workspace Web Edition (WWE), connected to the SIP Server T-Controller by issuing a TMonitorNextCall request. The request must contain:

- AttributeMonitorNextCallType, which defines the type of call supervision. Possible values are MonitorOneCall and MonitorAllCalls.
- AttributeExtensions/MonitorScope, which defines the scope of call supervision. Only **call** scope is supported and supervision is continued once started until the call is cleared.
- AttributeExtensions/MonitorMode, which defines the mode of call supervision. Only **silent** monitoring mode is supported. A supervisor can switch to **connect** mode after a call is established with an agent.

Supported supervision features

- A monitoring session can be started only for a call that arrives at the monitored Routing Point in a SIP Cluster node that is in the same data center (geo-location) as a supervisor owner node.
- Cross data center supervision is not supported. A monitoring session is not started if a call arrives at the other data center where a supervisor owner node is present.
- Intrusion is not supported. A monitoring session is not started for the calls that are already in queue when monitoring subscription is created.
- A monitoring session can be canceled by a T-Library client, such as WWE, by issuing a TCancelMonitoring request.
- One supervisor can monitor only one Routing Point or an agent.

Monitoring sessions for multiple supervisors on the same Routing Point

Multiple supervisors can monitor the same Routing Point. However, only one supervisor is allowed per call. If two supervisors monitor the same Routing Point, for the first call on the monitored Routing Point, SIP Server adds information on both supervisors in AttributeExtensions generated on the IVR observing Routing Point. The strategy tries each supervisor until a call is routed to one of them successfully. For the second call on the monitored Routing Point, SIP Server adds the supervisor that is not selected for the first call.

Each SIP Cluster node in the data center, where supervision is activated, maintains a queue of supervisors who are waiting to monitor a call on a Routing Point DN. Each call arriving at this Routing Point is given a list of available supervisors, and a strategy routes the call to the first available supervisor. If one supervisor fails to join the call, that supervisor is moved to the end of the line.

Enabling monitoring on a Routing Point DN

1. Create an observing Routing Point DN where a monitoring strategy will be loaded—for example, **6000**.
2. On the VoIP DN containing **service-type**=`sip-cluster-nodes`, set the **ivr-observing-routing-point** option to the Routing Point DN where a monitoring strategy is loaded. For example, **ivr-observing-routing-point**=`6000`.
3. Create a monitoring routing strategy. The routing strategy should check for each available supervisor from a list of supervisors provided by SIP Server and try routing to available supervisors until a call is successfully routed to one supervisor. If no supervisor is selected, the routing strategy should respond with TRouteCall (otherDN = <BLANK_VALUE>, RouteType = Reject). See a sample strategy.
4. Upload the strategy to the Routing Point DN **6000**.
5. Configure URS instances in SIP Cluster by following guidelines in URS configuration.
6. Create an application that will apply a treatment of playing the IVR menu to a call.

Sample call flow: Monitoring session for one supervisor with scope=call and type=OneCall

1. A supervisor logs in to the WWE desktop on Extension DN **2000** using AgentID= 'supervisor@onecloud.net'.
2. From the WWE desktop, the supervisor sends a TMonitorNextCall request containing AttributeOtherDN = 5000, which means that Routing Point DN **5000** is the monitoring target. Monitoring is started with the scope **call** and type **OneCall**.
3. SIP Server creates a monitoring subscription and sends a confirmation to the desktop using EventMonitoringNextCall.
4. SIP Server starts a monitoring session when a new call arrives at one of the SIP Cluster nodes in the data center where the owner of the supervisor desktop resides.
5. SIP Server transforms the inbound call to a conference by adding the IVR observing Routing Point to the call. SIP Server generates EventQueued on DN **6000**. This event contains the following key in the AttributeExtensions: 'Agents'='supervisor@onecloud.net'.
   - If a supervisor is not in a Ready state, URS distributes TRouteCall with the RouteReject type.
6. A routing strategy on DN **6000** routes the call to the supervisor by generating a TRouteCall request containing AttributeOtherDN = 2000.
7. SIP Server completes the conference and allows the supervisor to listen to how a caller is going through the IVR menu.
8. The supervisor desktop displays the monitoring information, such as the monitored DN number and its state.
9. The supervisor decides to continue monitoring the call even after the IVR stage is completed.
10. The call is routed to an agent.
11. The supervisor continues to monitor the call listening to how the agent talks to the caller.
12. SIP Server terminates the supervision subscription as the type was set to **OneCall**. The monitoring session won't be triggered for this supervisor when a new call arrives.

URS configuration

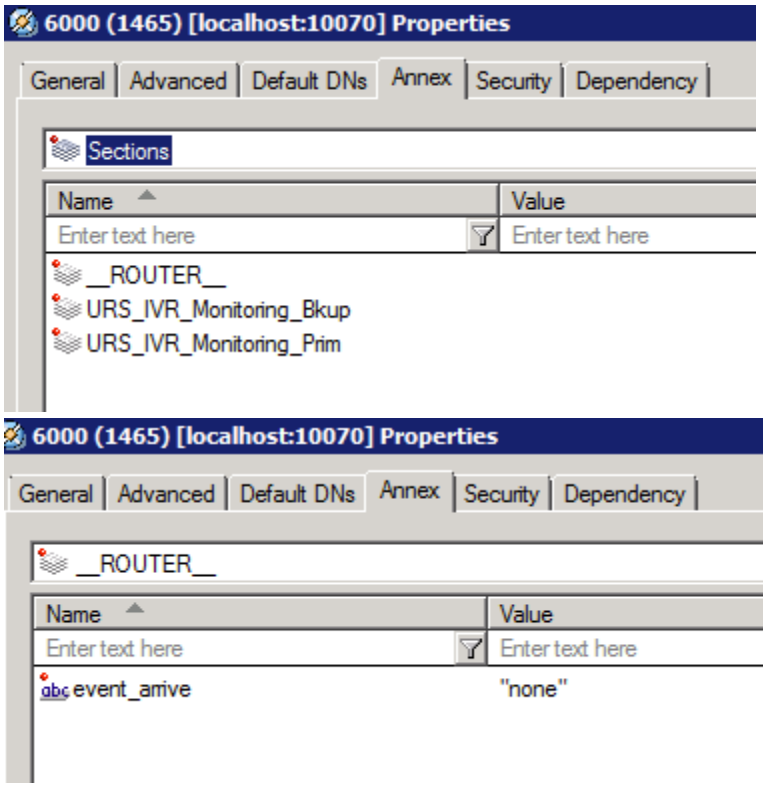
In this feature, SIP Server queues calls on two different Routing Points (a monitored Routing Point and an observing Routing Point specified in the **ivr-observing-routing-point** option) with the same ConnID at the same time. URS cannot run two strategies for the same call (the same ConnID) simultaneously. For this feature to work with URS, there must be one dedicated URS HA pair in each data center installed. This additional URS pair will monitor only the Routing Point that is configured in **ivr-observing-routing-point**. The other URS in the environment should not monitor the Routing Point specified in **ivr-observing-routing-point**. This new pair of URS instances must be connected to the routing Stat Server and to the default ports of all SIP Cluster nodes in this data center.

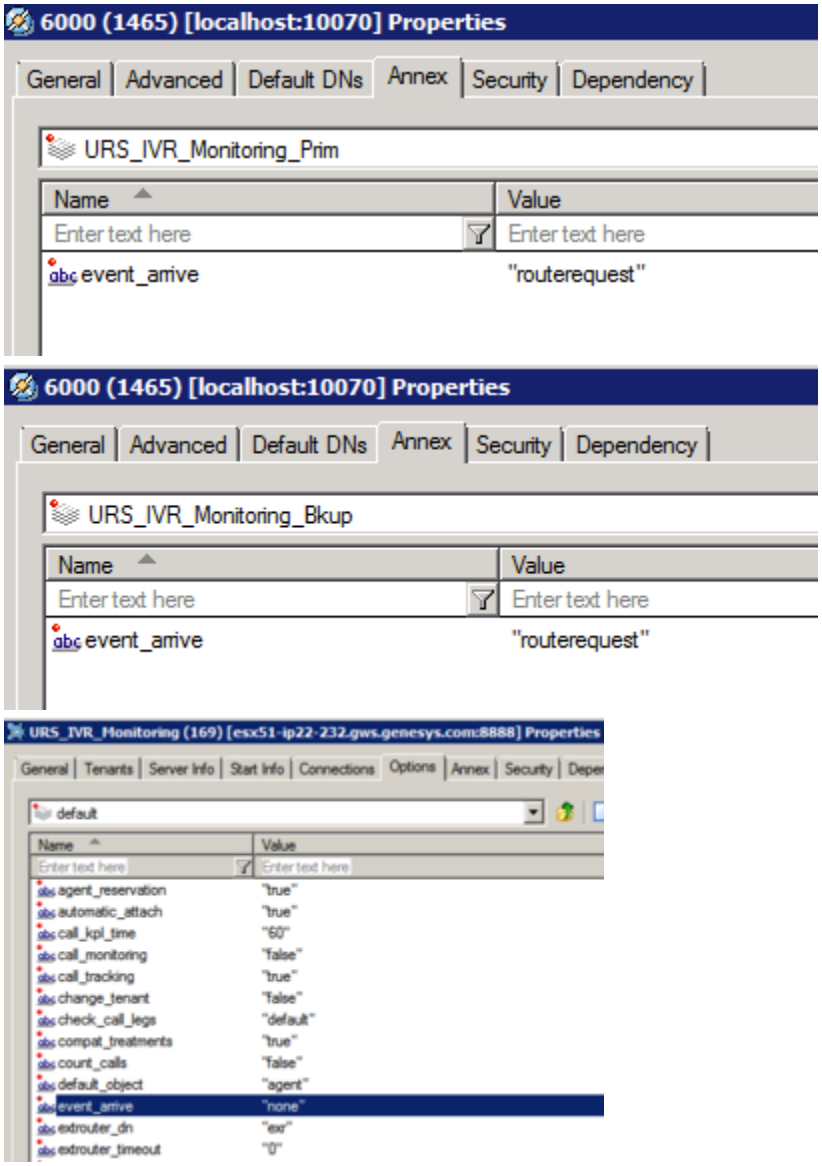
Complete the following configuration for URS applications to work with this feature:

- In the Annex tab of the Routing Point DN that is specified in the **ivr-observing-routing-point** option, create a section for each of the new URS applications (both primary and backup instances). In each section, add the **event_arrive** option and set it to `routerequest`. In addition, add another section with the name **__ROUTER__**. In that section, add the **event_arrive** option and set it to none. This is to make sure that an observing routing point is monitored by only the dedicated URS present in each data center.
- In the new URS applications, under the **default** section, set the **event_arrive** option to none.

Sample configuration for URS primary and backup instances

In this sample configuration, the following URS application names are used: **URS_IVR_Monitoring_Prim** and **URS_IVR_Monitoring_Bkup**.





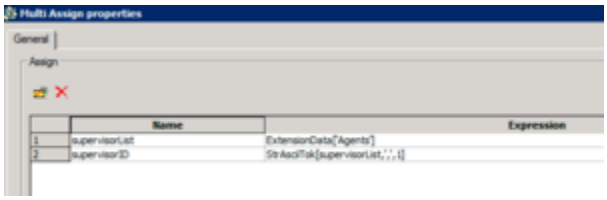
(Click the picture to expand)

Routing strategy sample

Here is an example of a routing strategy to be loaded on the IVR observing Routing Point:

(Click the picture to expand)

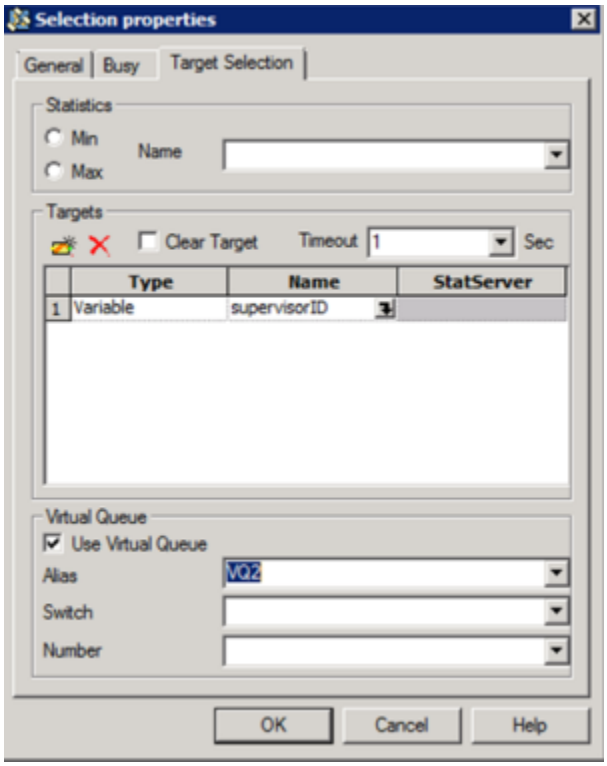
The sample strategy contains a Multi Assign object. A comma-separated list of supervisor IDs is retrieved from the EventRouteRequest AttributeExtensions by the ExtensionData function. From the list of supervisors, the first supervisor is taken.



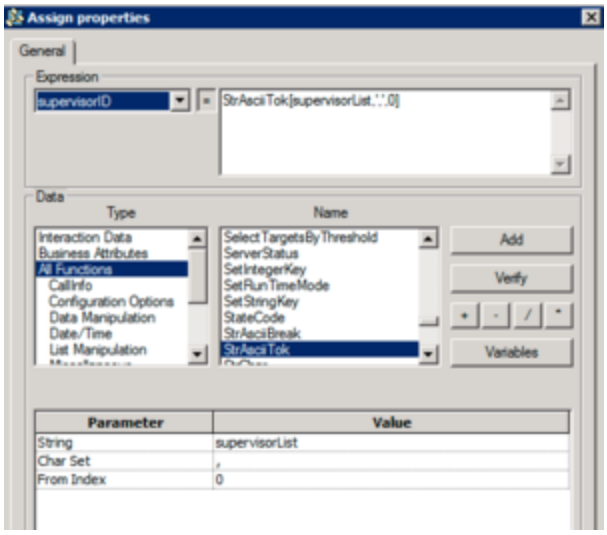
(Click the picture to expand)

If the value retrieved is not null, the strategy tries to route a call using a target selection block. If the call is routed to the supervisor successfully, the strategy is completed.

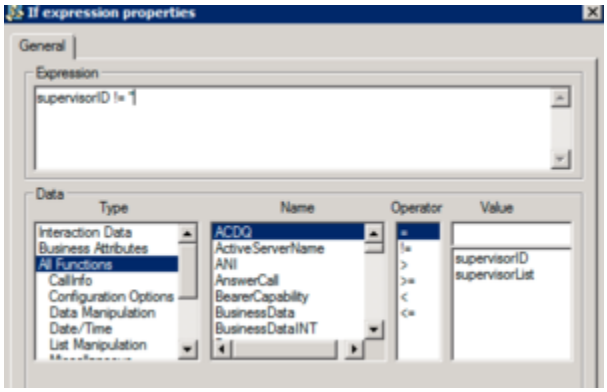
If the target selection block fails, the next supervisor ID is selected from the list and the routing is tried until a call is successfully routed.



(Click the picture to expand)

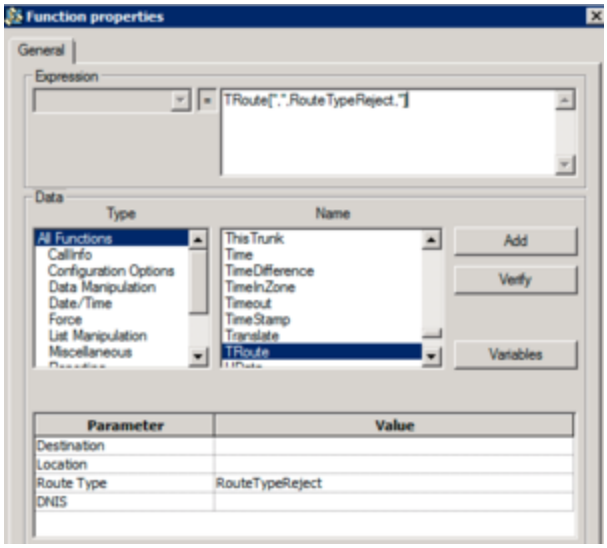


(Click the picture to expand)



(Click the picture to expand)

If no supervisor is available, the strategy rejects the call with the RouteReject type.



(Click the picture to expand)

Configuration option

The following configuration option is required to enable this functionality:

ivr-observing-routing-point
Default Value: No default value
Valid Values: The name of the Routing Point DN where a monitoring strategy is loaded
Changes Take Effect: Immediately

Specifies the name of the Routing Point DN where a monitoring strategy is loaded. This option is to be configured on the VoIP DN containing the **service-type** option set to `sip-cluster-nodes`.

External components configuration and dependencies

- For this feature, the **agent-reservation** option should be enabled by setting it either to `true` or `implicit` in the URS Application level in the **default** section.
- WWE should allow a supervisor to issue TMonitorNextCall for a Routing Point DN.

HA considerations

If a switchover occurs after a supervisor created a subscription for a Routing Point, a new Primary SIP Server maintains this subscription as active.

Feature limitations

- Only **call** scope is supported.
- Only **silent** supervision is supported. A supervisor can switch monitoring mode from **mute** to **connect**

and vice versa only after a call is established with an agent. Switching to **coach** mode is not supported.

- Intrusion is not supported; that is, a monitoring session will not be started for the calls that are already staying in queue when a monitoring subscription is created.