



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

GIS Deployment Guide

Introducing GIS

Contents

- 1 Introducing GIS
 - 1.1 About the Genesys Integration Server
 - 1.2 GIS Architecture
 - 1.3 System Requirements
 - 1.4 GIS Use-Case Scenario
 - 1.5 Licensing
 - 1.6 Backward Compatibility

Introducing GIS

This chapter describes Genesys Integration Server (GIS) architecture, the APIs and services that GIS exposes, and the basic requirements for deployment.

About the Genesys Integration Server

GIS is an XML server implemented as a web application. Developers use GIS to access the Genesys Framework and the Customer Interaction Management Platform by using one or more APIs to expose available SDK services.

Container and Client-Server Options

GIS is an XML server implemented as a web application that is embedded in a Tomcat, WebSphere, or WebLogic web container.

Important

This guide refers to the Tomcat deployment mode as a "stand-alone deployment." GIS 7.6 supports Tomcat versions 5.0.x and 5.5.x.

GIS is compatible with the Java 7 Platform, Enterprise Edition (J2EE). GIS is based on open Internet technologies and industry standards, such as eXtensible Markup Language (XML) and Simple Object Access Protocol (SOAP).

Important

In 7.6.6, GSAP protocol is discontinued

The GIS client interface conforms to Web Services Description Language (WSDL) specifications. It uses HTTP as the underlying message transport layer. Because GIS is based on common technologies, you can develop your client applications using one of several supported toolkits. For a list, see [GIS Built with Flexible Technologies](#).

For details, see the [Genesys Supported Operating Environment Reference Guide](#).

GIS Services as Represented in the Configuration Layer

To integrate available GIS services into the Configuration Layer of a Genesys environment, you need to map those services to Genesys Configuration Layer Application objects. The services then become Configuration Layer entries that expose Genesys Integration Server services. Services do not function

as servers. Each instance of Agent Interaction Service provides you with connections to the T-Servers, Interaction Server, and Universal Contact Server and each instance of Open Media Interaction Service provides you with connections to Interaction Service and Universal Contact Server.

Each GIS-exposed Genesys SDK Service is composed of three components:

- GIS, which presents the necessary APIs.
- Documentation that explains how to develop a client application that can successfully access the API. The documentation set for each SDK Service includes a *Developer's Guide* and an *API Reference*. The developer's guides also contain the code used in the code examples, accompanied by brief comments explaining the methods used, for instance, the Agent Interaction SDK Services Developer's Guide.
- Code examples, available on the [ISDK code samples page](#), consisting of functioning code examples in Java and C# that *exercise-related functionality and provide a model for how to write code to make your application perform specific functions*.

Important

All code examples are as accurate as possible. However, they are examples and are not tested or supported as fully functional code. Genesys does not guarantee that you can join the examples to create a complete application.

Session Service

Session Service—provides login, logout, and licensing functionality.

The Session Service provides an interface for login, logout, and licensing functions. Your client application must address the Session Service and receive licensing validation before it can communicate with the other services. The Session Service does not include a documentation component.

Configuration SDK Service

Configuration Service—provides an interface to Configuration Layer objects, enabling your client applications to access and modify Genesys configuration data.

The Configuration SDK Service enables you to view and modify configuration information in the Genesys Configuration Layer, using a locally developed, user-customized application.

Possible customizations include the following scenarios:

- A multi-site enterprise might provide access to configuration data for every tenant.
- Your client application might present data to a web page, which off-site supervisors and administrators can access.
- Your application might retrieve configuration updates based on time or change criteria.
- You might configure client applications or their users to access subsets of data, for instance, configuration data for only certain switches, queues, or agent groups.

- Your enterprise might use a centralized client application to retrieve, by way of the Internet, updated configuration information from multiple GIS servers deployed at different sites.
- Your third-party application can be integrated with Genesys Configuration to provide real-time synchronization, in one or both directions.

Examples of some of the above functionality are reproduced, with comments and suggestions for developing your client applications, in the Configuration SDK Web Services Developer's Guide.

Statistics SDK Service

Statistics Service—provides an interface to Stat Server statistics, granting your custom client applications access to Genesys real-time and historical data.

The Statistics SDK Service facilitates development of client applications that gather statistics about resource utilization, while introducing tools, such as filters, that enable a business-driven evaluation of contact-center operation and agent performance.

Users with a license only for the Statistics SDK Service can access read-only configuration information through the Configuration SDK Service using a limited selection of the Configuration Service methods.

Client applications using the Statistics SDK Service might perform such specialized functions as the following:

- In an enterprise with many T-Servers® and Stat Servers, provide access to statistical subsets for particular T-Servers, switches, or agent groups.
- Present data to a desktop application, to a wallboard display, or to a particular web page.
- Retrieve statistics on contact-center traffic, or employee productivity, based on time or change criteria.
- Access subsets of data, for instance, data that helps supervisors monitor agent performance or that enables managers to monitor campaign effectiveness.
- Retrieve, using the Internet, updated statistical information from GISs at multiple sites.

Examples of some of the above functionality are reproduced, with comments and suggestions for developing your client applications, in the Statistics SDK Web Services Developer's Guide.

Agent Interaction Services

Agent Interaction services—facilitates development of custom applications for interaction handling. For example, voice, e-mail, and/or chat interactions.

The Agent Interaction services enable you to develop applications for purposes like the following:

- Create a contact-center agent desktop application for Genesys software implementations.
- Integrate Genesys software with third-party software.
- Create other, specialized applications tailored to your needs. Typical usage scenarios include:
 - Managing agent login activity.
 - Handling e-mail interactions: sending, receiving, replying.
 - Handling voice interactions: calling, receiving, callback.
 - Handling chat interactions.

- Handling third-party media interactions.
- Handling outbound campaign participation.

Examples of some of the above functionality are reproduced, with comments and suggestions for developing your client applications, in the Agent Interaction SDK Java Developer's Guide. Typically, your applications will use the Agent Interaction services to establish connections to Genesys' Framework, Internet Contact Solution, and Outbound Contact Solution servers; to manage agent login activities; and to manage interactions.

Important

- Route Point and Route Service functionality are restricted within the Agent Interaction services, as deployed on GIS.
- Since 7.6.4, Agent Interaction services supports only one Stat Server connection

Open Media Interaction Services

Open Media Interaction services—enable you to build applications that manage third-party media interactions in the Genesys Framework. Such applications might submit third-party media interactions to Interaction Server; handle interactions' extensions through Interaction Server, using the External Service Protocol (ESP); manage third-party media interactions in the Universal Contact Server database; or manage the system's Customer Interaction Management queues. The Open Media Interaction services enable you to develop applications for purposes like the following:

- Maintaining a queued interface for Genesys media types and for third-party media.
- Getting business attributes and their values.
- Monitoring the changes in queues' state, and in associated interactions.
- Getting events on interactions in queues.
- Maintaining a media interface for third-party media types.
- Creating and managing third-party media interactions submitted to Interaction Server.
- Managing third-party media interactions in the Universal Contact Server database.
- Using the External Service Protocol (ESP) to handle interactions' extensions, through Interaction Server.

Examples of some of the above functionality are reproduced, with comments and suggestions for developing your client applications, in the Open Media Interaction SDK Services Developer's Guide.

GIS Architecture

The following architecture diagram shows the connections between GIS, the Genesys Framework, and your client application.

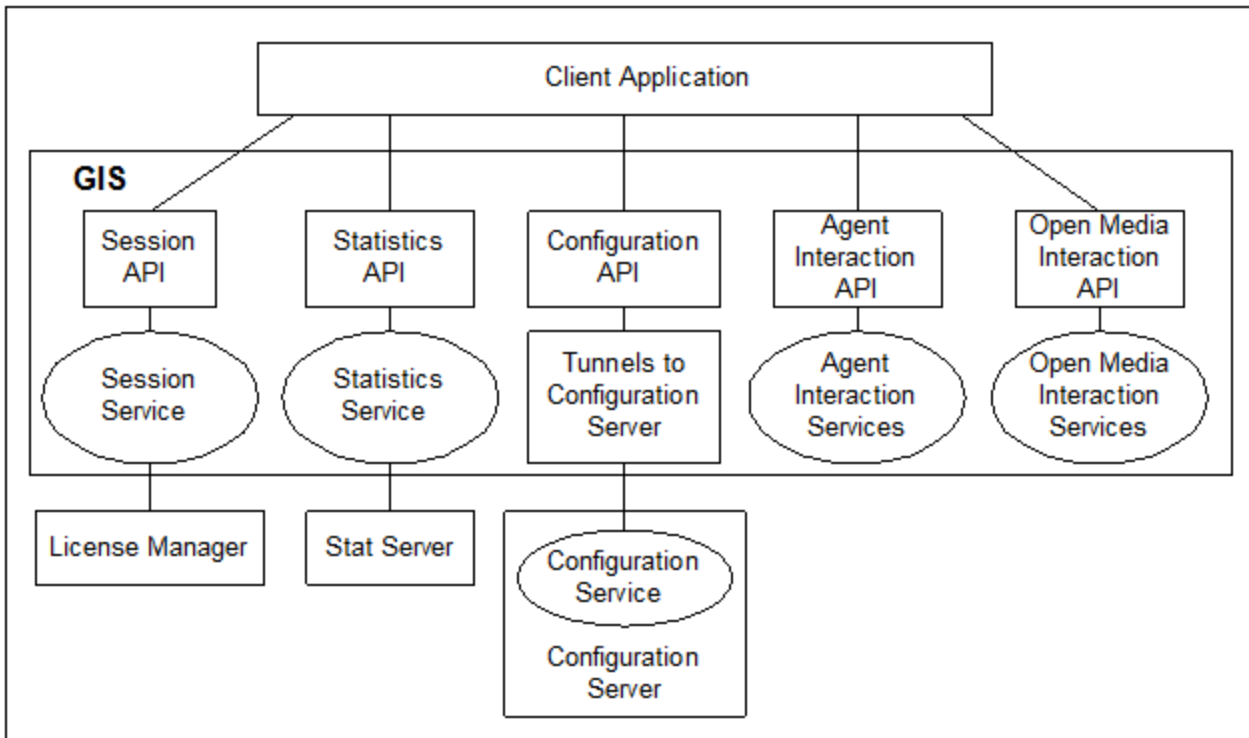
Important

- All the APIs and services shown in the diagram below]] are available to GIS:SOAP installations. Those labeled with an asterisk (*) are available only to GIS:SOAP installations.
- A single instance of GIS can support multiple client applications at the same time.

GIS works with the Genesys Framework in the following ways:

- GIS uses Configuration Server and License Manager for authentication and authorization.
- GIS accesses Configuration Server for connectivity information for Genesys Framework components.
- GIS supports integration with the Management Layer and is visible from Solution Control Interface.
- GIS communicates with Stat Server to retrieve statistical information that GIS presents through the Statistics Service.
- GIS communicates with Configuration Server's SOAP interface to access and modify configuration data that GIS presents through the Configuration Service.
- GIS communicates with the Agent Interaction Services to handle interactions on voice, e-mail, or chat media.

You can deploy single or multiple GIS instances that access the same or multiple Framework components. These connections are configured using Configuration Manager.



GIS Overall Architecture

GIS Built with Flexible Technologies

GIS client applications communicate with GIS using SOAP over HTTP only from release 7.6.6. For SOAP clients, communication conforms to a set of request/response operations that use XML as a basis for SOAP and WSDL definitions and behaviors.

If you are developing in Java, or in .NET-supported languages like Microsoft C#, Visual C++ .NET, or VB.NET, you can generate stubs or proxies that connect your application to GIS. However, Genesys recommends that you use the Genesys Integration Server Proxies/Libraries that Genesys provides (in both Java and .NET versions for the SOAP protocol). These proxies/libraries incorporate reconnect and defense mechanisms, as well as performance enhancements.

By deploying GIS as a SOAP connector, its reliance on HTTP as the underlying message transport layer means you can use existing, well-known communications and security functions in your applications. Only HTTP ports need to be opened. This permits minimal data transfer, which enables the use of standard ports and speeds up communications.

Supported Toolkits

Because of their XML-based nature and open architecture, the Statistics SDK Service and Configuration SDK Service enable developers to choose among various commercially available tools to build a SOAP-compliant client application that uses HTTP as the transport protocol. You should be able to use most major-market toolkits to develop successful client applications. The following have been tested and are officially supported:

- Microsoft .NET Framework SDK, version 2.0, 3.0, 3.5, 4.0 and 4.5 (<http://msdn.microsoft.com/>)

netframework/)

- Apache AXIS toolkit, version 1.4 (<http://xml.apache.org/axis/index.html>)

Adding a Web Server in Front of GIS

GIS resides in Tomcat, WebSphere, or WebLogic. If you plan to put an additional web server in front of GIS, see that web server's documentation for deployment instructions.

System Requirements

GIS is compatible with several hosts. For details about GIS system requirements, see the [Genesys Hardware Sizing Guide](#) . Generally, the GIS host should:

- Have at least 8 GBytes RAM.
- Have a CPU speed above 1.5 GHz, with 1.7 GHz as a minimum for Windows platforms.

Important

GIS software is multi-threaded and can take advantage of hosts with multiple CPUs.

- Have network access of at least 10 Mbps. However, Genesys highly recommends 100 Mbps or higher, in particular for SOAP deployments.

Install the Java Runtime Environment (JRE) or Java JDK, version 7 or 8:

- Windows and Solaris users should download the necessary software from the Oracle site at <http://www.oracle.com/technetwork/java/javase/downloads/index.html> .
- For Linux (Red Hat)
 - For REHL6:

Download and install Java from the Oracle site: [here](#).

Install also the 32 bits C++ package (which is not installed by default), required for the GIS installer package. To check the list of installed packages, use this command.

```
rpm -q compat-libstdc++-33
```

If the packages are not correctly installed, download and install the compat-libstdc++-33 package available [here](#)

To install the package, enter the following command at the prompt:

```
rpm -i compat-libstdc++-33-3.2.3-69.el6.i686
```

```
rpm -i compat-libstdc++-33-3.2.3-69.el6.x86_64
```

- For REHL7, check that the compatibility packages are installed:

```
rpm -q compat-libstdc++-33
```

If not, install them with the following procedure:

- Enable optional repo for yum:

```
yum-config-manager --enable rhel-7-server-optional-rpms
```

- Then install the following packages:

```
yum install compat-libstdc++-33-3.2.3-72.el7.i686.rpm
```

```
yum install compat-libstdc++-33-3.2.3-72.el7.x86_64
```

- AIX users should download the necessary software from <http://www.ibm.com/developerworks/java/jdk/aix/service.html>
- HP-UX users should download the necessary software from <https://h20392.www2.hp.com/portal/swdepot/displayProductInfo.do?productNumber=HPUXJAVAHOME>

To start GIS installation, run the `install.sh` script.

Important

Certain deployment scenarios require specific Java components:

- When deploying GIS using the Web Module option, install the full Java SDK, not just the JRE.
- When deploying GIS on the Solaris operating system, install JDK 7

For details, see the [Genesys Supported Operating Environment Reference Guide](#). Other deployment notes:

- The GIS installation package includes the recommended version of Apache AXIS and Tomcat. Genesys encourages you to use these versions to generate proxies (Tomcat 6 is supported).
- You can deploy GIS into a WebSphere or WebLogic web container.

Warning

In this release, deployment into WebLogic is supported on the Solaris platform only.

- The GIS directory tree consumes approximately 60 MB for SOAP deployments, with slight variations depending upon operating system.
- To integrate a stand-alone GIS instance with the Genesys Management Layer's Solution Control Server (SCS), you must install Local Control Agent (LCA), version 7.0.100.05 or higher, on the host that supports GIS.

Supported Operating Systems

For supported operating systems, see [Genesys Supported Operating Environment Reference Guide](#). This guide is updated regularly.

Important

Before installing GIS on a Red Hat Enterprise Linux 6 64-bit operating system, you must first install the Red Hat compatibility packages.

Sizing Guidelines

For sizing guidelines, refer to the Genesys Integration Server chapter in the [Genesys Hardware Sizing Guide](#), which you can access from the Technical Support website. This document is updated regularly.

Production vs. Development/Test Environments

You can deploy a simplified development/test environment using one of the following architectures:

- Install all required software on a single host. This includes GIS, DB Server, Configuration Server, Configuration Manager, and also all of the client development tools, including compilers, toolkits, and so on.
- Install all of the client development tools on one host and put all of the servers on another.
- Locate the client development tools and GIS on a single host, with the Genesys Framework resources on another host.

For more information, refer to the Genesys Integration Server chapter in the [Genesys Hardware Sizing Guide](#)

Under any deployment architecture, if you require the Statistics SDK Service, ensure that GIS has access to Stat Server.

Framework and Solutions Compatibility

Important

This section describes how some Genesys systems work together (interoperability).

For details, see the [Genesys Interoperability Guide](#).

Statistics SDK Service

The Genesys Statistics SDK Service is compatible with single-tenant or multi tenant Framework

versions 6.5 and higher. To manage configuration data, you must use either Configuration Server 7.x (which can be configured either as a master Configuration Server or as a proxy), or CS Proxy 6.5.

Configuration SDK Service

The Configuration SDK Service requires either Configuration Server 7.x (which can be configured as either a master Configuration Server or a proxy) or CS Proxy 6.5. The Configuration SDK Service is compatible with single tenant or multi-tenant versions of the Genesys Framework.

Agent Interaction Service

The Agent Interaction SOA (Service Oriented Architecture) is compatible with Framework 7.0 and with the following Genesys products:

- Outbound Contact Solution (OCS): 7.x
- Voice CallBack: 7.x
- Multimedia or Multi-Channel Routing (MCR): 7.x

Important

For basic Open Media features, you must have Multi-Channel Routing 7.1.x, or Multimedia 7.2.x or higher. For complete Open Media features (including Queued Interaction Layer and Media Interaction Layer), Multimedia 7.2.x or higher is mandatory.

Open Media Interaction Service

The Open Media Interaction Service is compatible with Framework 7.1 or higher, and with Multi-Channel Routing 7.1.x or Multimedia 7.2.x and higher.

Management Layer Support

If you are using GIS in a Tomcat web container, you can start and stop GIS using Solution Control Interface (SCI).

To use the Management Layer, you must be running LCA (Local Control Agent) 7.0.1 or higher.

Important

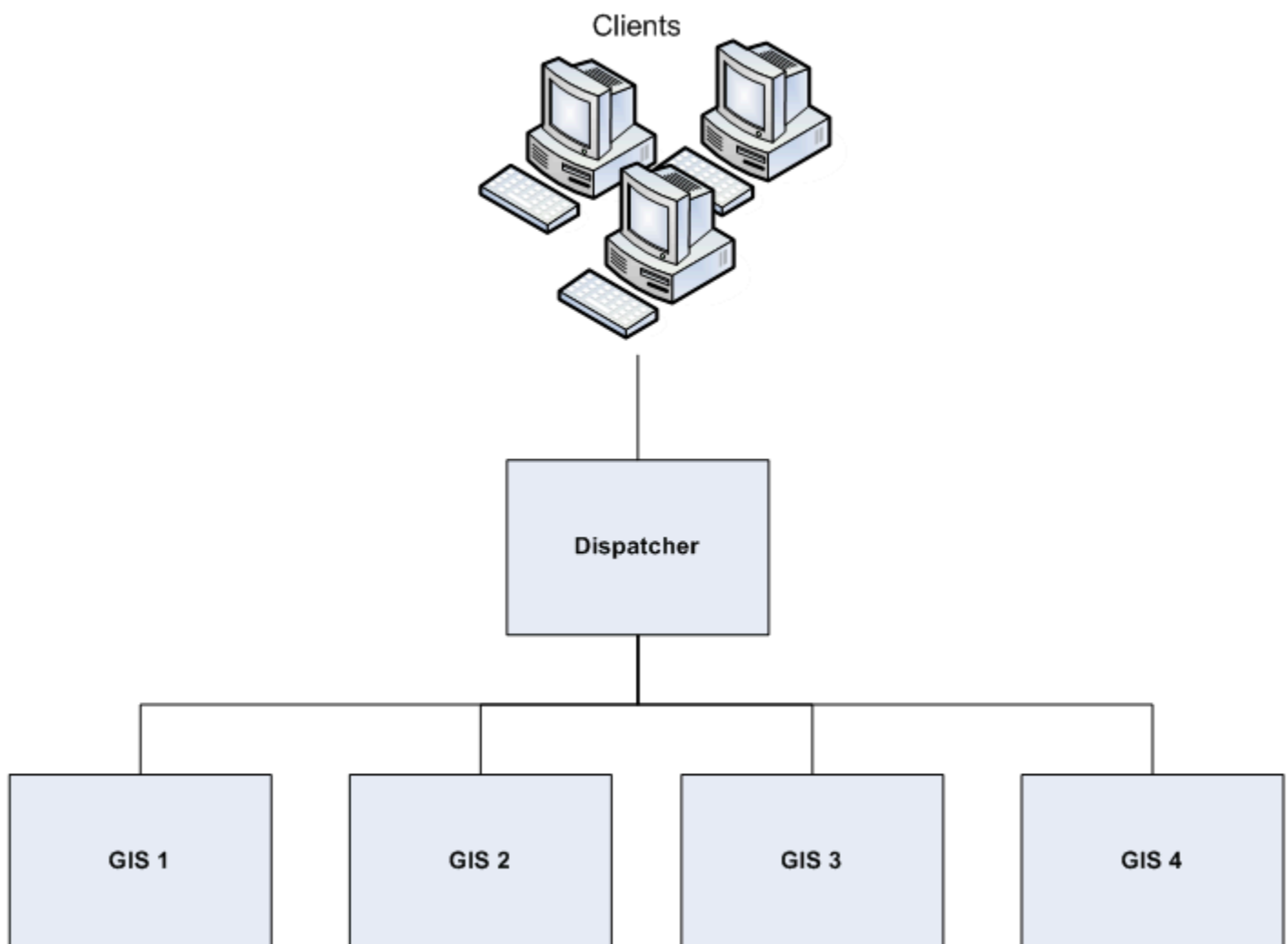
You cannot use the Management Layer to control GIS if you are running GIS in a WebSphere or WebLogic container.

GIS Use-Case Scenario

This section provides details from one GIS deployment that Genesys implemented at a customer site. The design offers a highly available, robust, and redundant architecture for two-thousand agents.

GIS:SOAP 7.6 - Clustering

The GIS:SOAP 7.6 HA with Load Balancing deployment is suitable for a highly available, robust, redundant deployment scenario (See [Example from a Load Balanced and Highly Available Environment](#), which details an actual GIS Deployment). The implementation consists of multiple instances of GIS in a cluster, [as shown in the following figure](#).



GIS SOAP 7.6 HA deployment using a GIS cluster

With this deployment, if the current client-GIS connection fails:

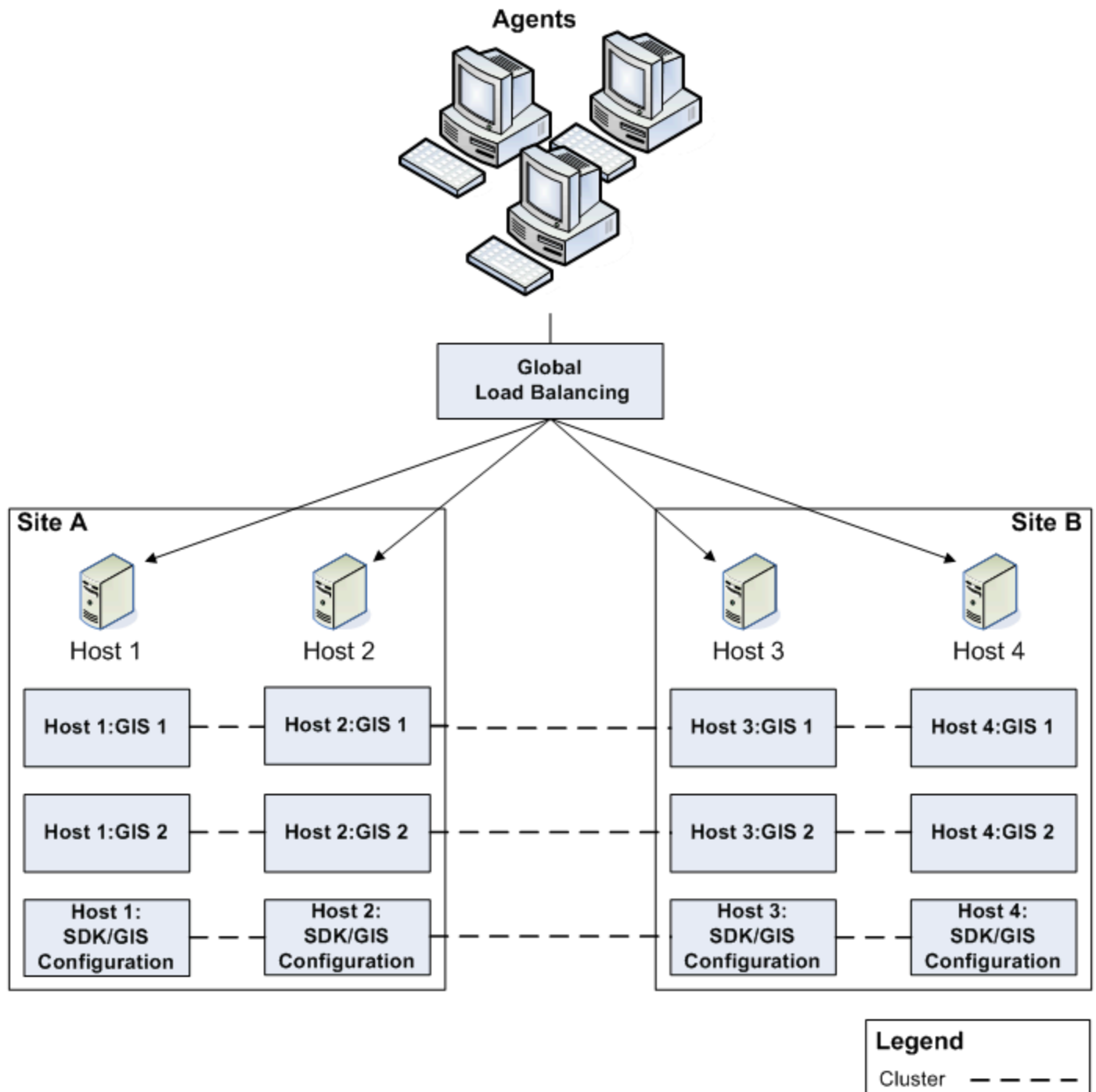
- The client application is automatically switched (by the dispatcher) to one of the other GIS nodes in the cluster.
- The session data is maintained by the cluster cache. Therefore, a database is not required.
- The agent session data is maintained. The agent does not need to log in again.

Important

For more information about this deployment, see [Deployment for High Availability with Load Balancing](#).

Example from a Load Balanced and Highly Available Environment

The following figure shows a multi-site, multi-host, multi-cluster, high availability environment that supports two-thousand (2000) agents.



Multi-site GIS cluster with HA

The environment includes:

- Agents
- Global Load Balancer (GLB)

- Multiple Sites (Site A and Site B)
- Multiple Host Machines (Hosts 1, Host 2, Host 3, and Host 4)
- Multiple GIS Instances (GIS 1, GIS 2, and SDK/GIS Configuration)
- Multiple GIS Clusters (GIS-cluster 1, GIS-cluster 2, and SDK/GIS Configuration cluster)

Agents

This environment supports a load of up to 2000 agents. The agent load is distributed over multiple sites and multiple hosts by GLB.

Global Load Balancer (GLB)

The GLB has a multi-dimensional configuration that distributes the agent load:

- Between two sites (four hosts).
- Across three GIS clusters.

To do this GLB is configured with three HTTP addresses (one per GIS cluster). When a request is received from an agent, a session is opened to one GIS cluster (only one!). Then, the sessions will be balanced between the four GIS instances in the cluster. For example, if one of the GIS instances becomes unavailable, the session is managed on the remaining instances in the cluster.

Important

GLB does not balance sessions across multiple GIS clusters. Once a session has been opened on a GIS cluster, it remains on that cluster.

Multiple Sites (Site A and Site B)

This sample environment is designed for multiple sites (Site A and Site B). This offers excellent availability. For example, if one site becomes unavailable, agents can connect to the second site and continue their work.

Multiple Host Machines (Host 1, Host 2, Host 3, and Host 4)

This design includes four host machines (two hosts per site). Each host has:

- Three GIS instances (GIS 1, GIS 2, and SDK/GIS Configuration)
- Two (2) GB for each GIS.

If a host or an application server on a host (Websphere) becomes unavailable, all agents connected to that host are redirected to another host by GLB.

Multiple GIS Instances (GIS 1, GIS 2, and SDK/GIS Configuration)

This sample environment includes three GIS instances per host:

- GIS 1—each instance of GIS 1 is configured to support up to 500 agents. It is the primary GIS on a host, and it is in cluster with the GIS 1s on the other host machines.
- GIS 2—each instance of GIS 2 is configured to support up to 500 agents. It is the backup instance on a host, and it is in cluster with the GIS 2s on the other host machines.
- SDK/GIS Configuration—supports ongoing configuration and any further SDK customization needed.

Multiple GIS Clusters (GIS 1 Cluster, GIS 2 Cluster, and SDK/GIS Configuration Cluster)

This sample environment includes three clusters:

- A cluster with four GIS 1s
- A cluster with four GIS 2s
- A cluster with four SDK/GIS Configurations

Typically, GLB directs agents to the GIS 1 Cluster. This cluster is replicated in the GIS 2 Cluster. However, GLB directs requests to the GIS 2 Cluster in the following circumstances:

- If the entire GIS 1 Cluster becomes unavailable (stops responding), agents connected to GIS 1 Cluster are redirected to GIS 2 Cluster.

If three-of-four host machines become unavailable (stop responding), agents are directed to either a GIS 1 or GIS 2 on the remaining host.

Licensing

GIS manages the license control for the Statistics, Configuration, and Interaction SDK Services, and manages the number of sessions allowed for each interface. GIS 7.6 requires License Manager 8.3, which is available on a separate CD.

When you purchase a license for a GIS SDK Service, you receive license keys for GIS, the appropriate SDK Service, and the Configuration Server Proxy interface (which was packaged separately in 6.x releases as CS Proxy).

Important

Both the Statistics and Configuration SDK Services require the Configuration Server Proxy license key to access configuration information. This includes the read-only configuration information for users who are purchasing only the Statistics SDK Service. If you are upgrading to GIS 7.6 from Genesys Interface Server 7.1 or earlier, you must migrate your existing license keys. For details, see the [Genesys Licensing](#)

Guide.

For more information on licensing, see:

- The [Genesys Licensing Guide](#) for a detailed discussion of Genesys product licensing and using License Manager.
- [Configuring the License Section of the Option Tab for GIS:SOAP](#), which contains instructions on how to configure license information.
- The *Developer's Guide* for each SDK Service. These explain how to configure your login message so that the client application is authorized to communicate with the correct SDK Service.

Backward Compatibility

GIS 7.6 servers are compatible with client applications written using Genesys 7.6, 7.5, 7.2 or 7.1 SDKs (Software Development Kits). However, features newly introduced in 7.6 will typically be available only if you also upgrade your client applications to incorporate new 7.6 SDK components.

Warning

The predecessor products, Genesys Interface Server 7.1 and Genesys .NET Server 7.1, are not compatible with client applications built using the Genesys 7.6 SDKs.

If you are upgrading from Genesys .NET Server 7.1, GIS 7.6 supports your existing client applications by exposing new versions of the same Genesys .NET services. You simply need to connect your client applications to GIS 7.6 instead of .NET Server 7.1.

If you are upgrading from Genesys Interface Server 7.1, GIS 7.6 supports your existing client applications by exposing new versions of all the same Genesys services.

With GIS release 7.6.6 and developments in .Net, while APIs have been preserved, proxies shipped for .NET have been migrated from 1.1 to 2.0. This requires customer applications to be recompiled against the new targeted framework (2.0 or higher).