



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.

# Platform SDK Developer's Guide

Introducing the Platform SDK

12/13/2025

---

## Contents

- 1 Introducing the Platform SDK
  - 1.1 The Protocols
  - 1.2 The Application Blocks
  - 1.3 The Components

# Introducing the Platform SDK

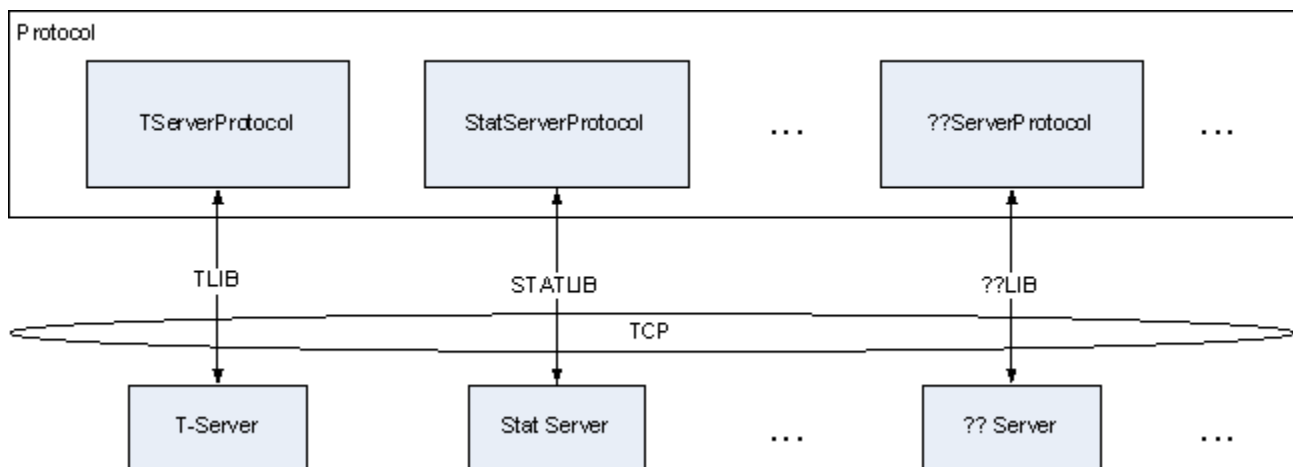
The Platform SDK exposes the protocols of Genesys servers as an API. This means you can write .NET and Java applications that communicate with these servers in their native protocols.

You can use the Platform SDK to do two main things:

- Establish and maintain a connection to each Genesys server used by your application
- Send and receive messages to and from each of these Genesys servers

In addition to enabling these two basic functions, the Platform SDK ships with *application blocks*, which have been built on top of the Platform SDK in order to provide simple yet high-performance ways to do things like configuring warm standby settings for your connections and working with configuration objects.

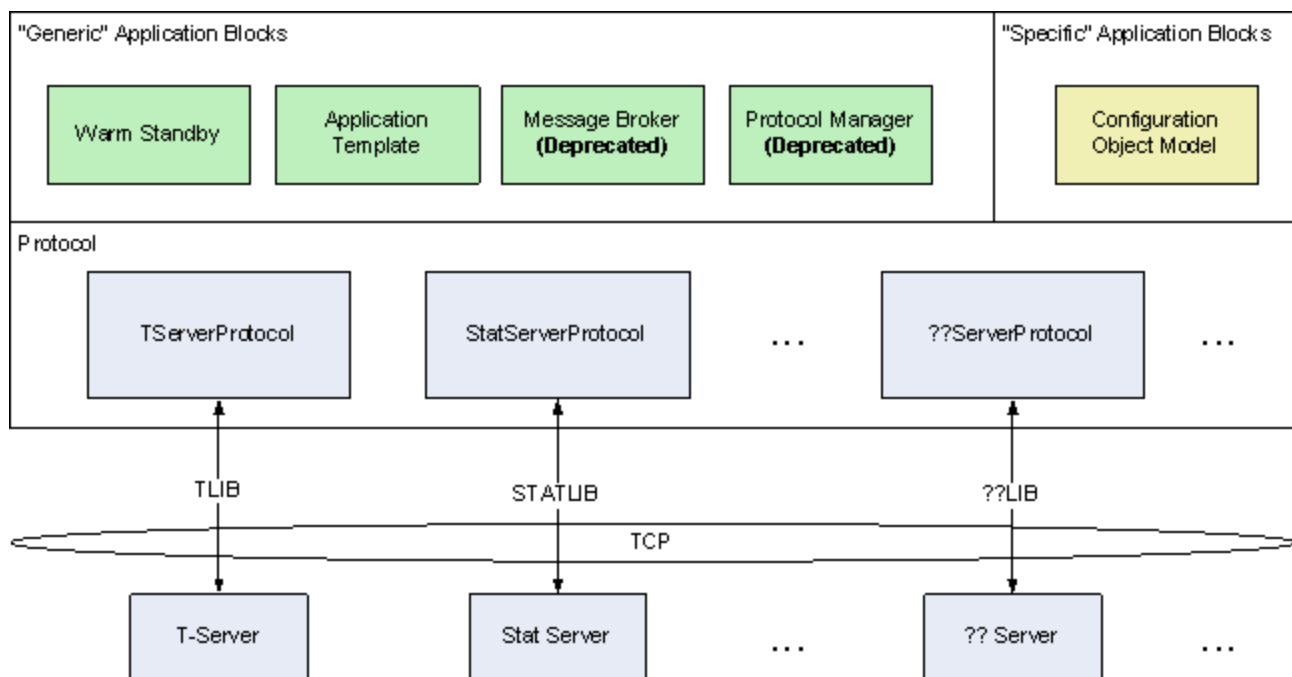
The following image shows the relationship between the Platform SDK protocol objects and the servers each of them connects with.



Each protocol object subclasses `ClientChannel`, which in turn subclasses `DuplexChannel` and implements the `Protocol` interface. This means they all share a common interface to the Genesys servers. The protocol objects communicate with the corresponding Genesys servers over a TCP connection, with each one using the native protocol of the server it connects with. For example, the `TServerProtocol` object communicates over TCP with a `T-Server`, using the `TLIB` protocol that is native to the `T-Server`.

As mentioned above, the Platform SDK also includes reusable production-quality application blocks that can be dropped into your code to provide simple yet high-performance ways to carry out important functions that are commonly needed by applications that communicate with Genesys servers.

As shown below, there are two main types of application blocks: generic and specific.



Generic application blocks provide functionality that is useful for a broad range of applications, such as configuring connection and warm standby settings. These application blocks are recommended for use in most development. Specific application blocks are only beneficial for certain types of applications. For example, the Configuration Object Model application block makes it easy to work with objects in the Genesys Configuration Layer and is only required when you are writing an application that requires this functionality.

Finally, the Platform SDK includes additional components designed to make development of custom applications easier. These components offer support for useful features such as customized logging or switch abstraction.

## The Protocols

The Platform SDK is divided into separate "protocols." Each component works with one or more of Genesys servers.

The following table shows the servers each of the Platform SDK protocols connects with, and gives the names of the native protocols that are used to communicate with each server.

Platform SDK Protocol Name	Genesys Servers	Native Protocols
Configuration Platform SDK	Configuration Server	CFGLIB
Contacts Platform SDK	Universal Contact Server	UCS Protocol
Management Platform SDK	<ul style="list-style-type: none"> <li>Message Server</li> <li>Solution Control Server</li> </ul>	<ul style="list-style-type: none"> <li>GMESSAGELIB</li> <li>SCSLIB</li> </ul>

Platform SDK Protocol Name	Genesys Servers	Native Protocols
	<ul style="list-style-type: none"><li>Local Control Agent</li></ul>	<ul style="list-style-type: none"><li>LCALIB</li></ul>
Open Media Platform SDK	Interaction Server	ITX, ESP
Outbound Contact Platform SDK	Outbound Contact Server	<ul style="list-style-type: none"><li>CMLIB</li><li>OCS-Desktop Protocol</li></ul>
Routing Platform SDK	<ul style="list-style-type: none"><li>Custom Server</li><li>Universal Routing Server</li></ul>	<ul style="list-style-type: none"><li>Custom Server Protocol</li><li>Routing Server Protocol</li></ul>
Statistics Platform SDK	Stat Server	STATLIB
Voice Platform SDK	T-Servers	<ul style="list-style-type: none"><li>TLIB</li><li>Preview Interaction Protocol</li></ul>
Web Media Platform SDK	<ul style="list-style-type: none"><li>Chat Server</li><li>E-Mail Server Java</li><li>Callback Server</li></ul>	<ul style="list-style-type: none"><li>MCR Chat Lib</li><li>MCR E-Mail Lib</li><li>MCR Callback Lib</li><li>ESP E-Mail Lib</li></ul>

### Configuration Platform SDK

The Configuration Platform SDK enables you to build applications that use the services of the Genesys Configuration Server. This allows these applications to either query on objects in the Configuration Layer of your Genesys environment or to add, modify, and delete information about those objects, while taking advantage of an environment in which Configuration Server carries out several important administrative functions.

### Contacts Platform SDK

The Contacts Platform SDK allows you to build applications that view, or interact with, the contact information for your contact center. This SDK accesses information directly from Universal Contact Server, allowing you to design applications that access contact information when dealing with multimedia interactions such as chat or email, for example.

### Management Platform SDK

The Management Platform SDK enables you to write applications that interact with Message Server, Solution Control Server, and Local Control Agents.

### Open Media Platform SDK

With the Open Media Platform SDK, you can build client applications that feed open media interactions into your Genesys environment, or applications that act as custom media servers to

perform external service processing (ESP) on interactions that have already entered it.

### **Outbound Contact Platform SDK**

The Outbound Contact Platform SDK can be used to build applications that allow you to manage outbound campaigns.

### **Routing Platform SDK**

The Routing Platform SDK allows you to write Java and .NET applications that combine logic from your custom application with the router-based logic of URS, in order to solve many common interaction-related tasks.

### **Statistics Platform SDK**

With the Statistics Platform SDK, you can build applications that use the services of Stat Server in order to solicit and monitor statistics from your Genesys environment.

Stat Server tracks information about customer interaction networks (contact center, enterprise-wide, or multi-enterprise telephony and computer networks). It also converts the data accumulated for directory numbers (DNs), agents, agent groups, and non-telephony-specific object types, such as email and chat sessions, into statistically useful information.

### **Voice Platform SDK**

The Voice Platform SDK enables you to design applications that monitor and handle voice interactions from a traditional or IP-based telephony device.

### **Web Media Platform SDK**

The Web Media Platform SDK can be used to build applications that interact with Chat Server, E-Mail Server Java, and Callback Server through a web server interface.

## The Application Blocks

### **Important**

These application blocks are reusable production-quality components, designed using industry best practices and provided with source code so they can be used "as is," extended, or tailored as necessary.

Please see the License Agreement for details.

Genesys application blocks are reusable production-quality components that provide specific functionality needed by a broad range of Genesys customers. They have been designed using industry best practices and provided with source code so they can be used "as is," extended, or tailored if you need to. Please see the License Agreement for details.

### **Application Template Application Block**

---

The [Application Template Application Block](#) provides a way to read configuration options for applications in Genesys Administrator and to configure Platform SDK protocols. It also allows standard connection settings (including ADDP or TLS details) to be retrieved from Configuration Server, and helps with common features like setting up WarmStandby or assigning message filters.

### **Configuration Object Model Application Block**

The [Configuration Object Model \(COM\) Application Block](#) provides a consistent and intuitive object model for applications that need to work with Configuration Server objects. Use the COM Application Block when you need to create, update, or delete Configuration Layer Objects.

### **Warm Standby Application Block**

You can use the [Warm Standby Application Block](#) to switch to a backup server in case your primary server fails, in cases where you do not need to guarantee the integrity of existing interactions.

## Deprecated Content

The application blocks listed in this section are considered legacy products. Documentation is still provided for backwards compatibility, but new development should not use these application blocks.

### **Message Broker Application Block (deprecated)**

The [Message Broker Application Block](#) makes it easy for your applications to handle events in an efficient way.

Deprecated with release 8.1.1. If you have existing applications that use the Message Broker Application Block, refer to the [migration](#) article for details on how to update your code. New applications should use the improved [message handling](#) capability now included in Platform SDK instead.

### **Protocol Manager Application Block (deprecated)**

The [Protocol Manager Application Block](#) allows for simplified communication with more than one server. It takes care of opening and closing connections to many different servers, as well as handling the reconfiguration of high availability connections.

Deprecated with release 8.1.1. If you have existing applications that use the Protocol Manager Application Block, refer to the [migration](#) article for details on how to update your code. New applications should use the improved ability of Platform SDK to [connect to servers](#) instead.

## The Components

Additional components are included to provide useful functionality for creating custom applications with the Platform SDK, even if that doesn't necessarily involve communicating with Genesys servers.

### **Platform SDK Log Library**

The Platform SDK Log Library presents an easy-to-use API for logging messages in custom-built applications.