



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.

Genesys Events and Models Reference

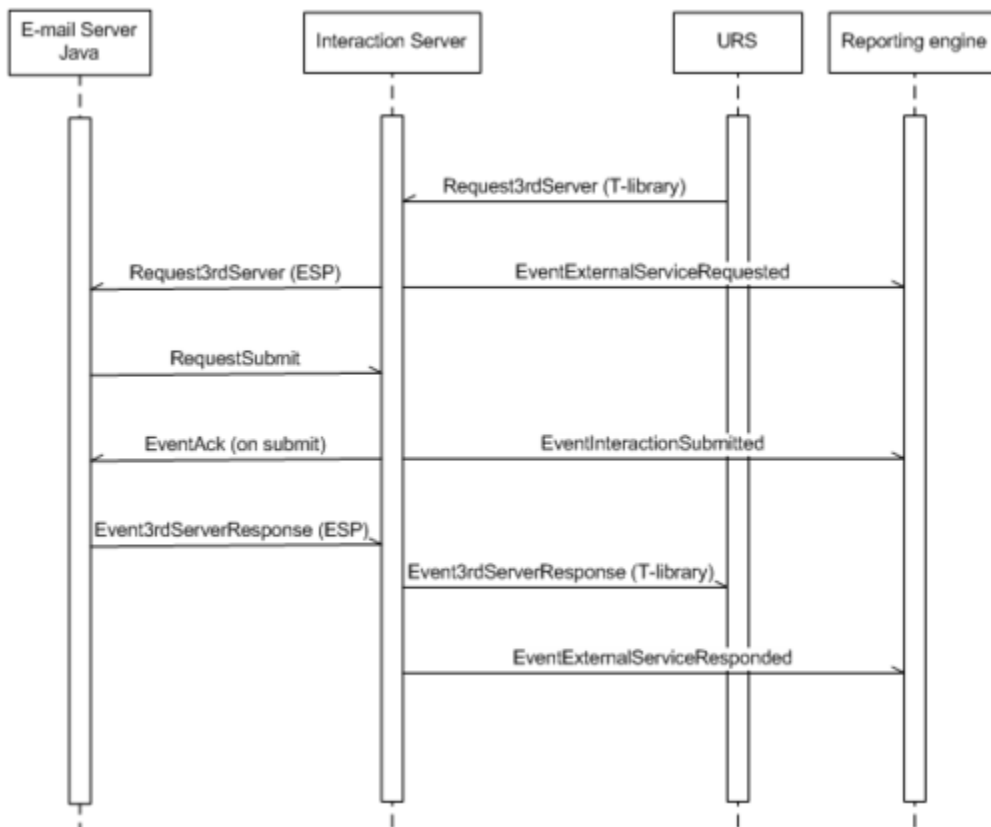
[Invoke Autoresponse](#)

Invoke Autoresponse

This set of models illustrates the following scenario:

1. URS, triggered by an Autoresponse strategy object, generates a request for E-mail Server Java to generate a new autoresponse interaction. It sends this request to Interaction Server using Request3rdServer.
2. Interaction Server relays the request to E-mail Server Java using the same Request3rdServer. It also relays the content of the request to the reporting engine using EventExternalServiceRequested.
3. E-mail Server Java generates a new Autoresponse interaction and submits it to Interaction Server using RequestSubmit. When Interaction Server acknowledges the submission, E-mail Server Java sends Event3rdServerResponse, which Interaction Server relays to URS. Interaction Server also relays the content of that event to the reporting engine using EventExternalServiceResponded.

This model, which is not further divided into phases, is shown below.



Invoke Autoresponse

This phase uses the messages shown in the following table:

Messages in Invoke Autoresponse

Message	Protocol
Event3rdServerResponse	ESP
EventAck	Interaction Management
EventExternalServiceRequested	Reporting
EventExternalServiceResponded	Reporting
EventInteractionSubmitted	Reporting

In this scenario, URS uses Interaction Server as an intermediary to communicate with E-mail Server Java, which in this case is called a *third-party server*. Another example of such a third-party server is Classification Server, with which URS must similarly communicate when a strategy includes a Classify object. To relay these messages to third-party servers, Interaction Server uses the T-Library messages RequestPrivateService and EventPrivateInfo, with special content in their extensions and user_data attributes. With that special content, these messages make up a small *External Services Protocol* (ESP), as shown in the following table:

ESP Messages

T-Library Message	ESP Message	Description
EventPrivateInfo	Event3rdServerResponse	Returns results of third-party server's operation
	Event3rdServerFault	Contains information about failure of third-party server's operation

ESP is not further described in this manual. However, much of the relevant content of Request3rdServer and Event3rdServerResponse is repeated in the attributes of the reporting protocol events EventExternalServiceRequested and EventExternalServiceResponded.

Components that understand ESP are called *ESP servers*. Classification Server is an ESP server. E-mail Server Java functions both as a media server (processing Interaction Management Protocol messages) and as an ESP server, processing ESP messages. You can also create custom ESP servers using the Genesys Open Media Platform SDK.