intelligent Workload Distribution 8.0

WebSphere MQ Capture Adapter

Reference Guide

About Genesys

Alcatel-Lucent's Genesys solutions feature leading software that manages customer interactions over phone, Web, and mobile devices. The Genesys software suite handles customer conversations across multiple channels and resources—self-service, assisted-service, and proactive outreach—fulfilling customer requests and optimizing customer care goals while efficiently using resources. Genesys software directs more than 100 million customer interactions every day for 4000 companies and government agencies in 80 countries. These companies and agencies leverage their entire organization, from the contact center to the back office, while dynamically engaging their customers. Go to www.genesyslab.com for more information.

Each product has its own documentation for online viewing at the Genesys Technical Support website or on the Documentation Library DVD, which is available from Genesys upon request. For more information, contact your sales representative.

Notice

Although reasonable effort is made to ensure that the information in this document is complete and accurate at the time of release, Genesys Telecommunications Laboratories, Inc., cannot assume responsibility for any existing errors. Changes and/or corrections to the information contained in this document may be incorporated in future versions.

Your Responsibility for Your System's Security

You are responsible for the security of your system. Product administration to prevent unauthorized use is your responsibility. Your system administrator should read all documents provided with this product to fully understand the features available that reduce your risk of incurring charges for unlicensed use of Genesys products.

Trademarks

Genesys, the Genesys logo, and T-Server are registered trademarks of Genesys Telecommunications Laboratories, Inc. All other trademarks and trade names referred to in this document are the property of other companies. The Crystal monospace font is used by permission of Software Renovation Corporation, www.SoftwareRenovation.com.

Technical Support from VARs

If you have purchased support from a value-added reseller (VAR), please contact the VAR for technical support.

Technical Support from Genesys

If you have purchased support directly from Genesys, please contact Genesys Technical Support at the regional numbers provided on page 9. For complete contact information and procedures, refer to the ***Genesys Technical Support Guide***.

Ordering and Licensing Information

Complete information on ordering and licensing Genesys products can be found in the ***Genesys Licensing Guide.***

Released by

Genesys Telecommunications Laboratories, Inc. www.genesyslab.com

Document Version: 80iwd_ref_mq-ca_08-2010_v8.0.001.00

Table of Contents

List of Procedures

Preface

Welcome to the *intelligent Workload Distribution 8.0 WebSphere MQ Capture AdapterReference Guide.* This document provides a detailed description of the intelligent Workload Distribution (iWD) interface to the IBM WebSphere MQ service bus.

This document is valid only for the 8.0 release of this product.

Note: For versions of this document created for other releases of this product, visit the Genesys Technical Support website, or request the Documentation Library DVD, which you can order by e-mail from Genesys Order Management at orderman@genesyslab.com.

This preface contains the following sections:

For information about related resources and about the conventions that are used in this document, see the supplementary material starting on page 43.

About intelligent Workload Distribution

iWD is an application that works with the Genesys Customer Interaction Management (CIM) Platform to distribute tasks to the resources best suited to handle them. It is a collection of software components for:

- Capturing tasks from various enterprise work sources.
- Applying business rules to classify, prioritize, and reprioritize the tasks.
- Routing the tasks to agents or knowledge workers in the enterprise.
- Monitoring and reporting on the intra-day and historical status of the tasks and the task handling.

iWD creates an enterprise-wide, or “Global Task List” that is centrally managed and prioritized. As such, it provides visibility for business analysts into the backlog of tasks yet to be completed, as well as the status of in-progress and completed tasks.

iWD provides a user interface designed specifically for business users, giving them not only access to the Global Task List, but also to a user interface that allows them to author business rules that describe the policies of the enterprise. For example, the business rules can be used to determine what priority and due date should be given to a task with a specific set of attributes.

Intended Audience

This document is intended for architects and developers who want to implement a project that leverages iWD via the WebSphere MQ interface. It has been written with the assumption that you have a basic understanding of:

- intelligent Workload Distribution (iWD) and WebSphere MQ messaging services.
- Network design and operation
- Your own network configurations

Making Comments on This Document

If you especially like or dislike anything about this document, feel free to e-mail your comments to Techpubs.webadmin@genesyslab.com.

You can comment on what you regard as specific errors or omissions, and on the accuracy, organization, subject matter, or completeness of this document. Please limit your comments to the scope of this document only and to the way in which the information is presented. Contact your Genesys Account Representative or Genesys Technical Support if you have suggestions about the product itself.

When you send us comments, you grant Genesys a nonexclusive right to use or distribute your comments in any way it believes appropriate, without incurring any obligation to you.

Contacting Genesys Technical Support

If you have purchased support directly from Genesys, contact Genesys Technical Support at the following regional numbers:

Region	Telephone	E-Mail
North and Latin America	+888-369-5555 (toll-free) +506-674-6767	support@genesyslab.com
Europe, Middle East, and Africa	+44-(0)-1276-45-7002	support@genesyslab.co.uk
Asia Pacific	+61-7-3368-6868	support@genesyslab.com.au
Malaysia	1-800-814-472 (toll-free) +61-7-3368-6868	support@genesyslab.com.au
India	000-800-100-7136 (toll-free) +91-(022)-3918-0537	support@genesyslab.com.au
Japan	+81-3-6361-8950	support@genesyslab.co.jp
Before contacting technical support, refer to the *Genesys Technical Support Guide* for complete contact information and procedures.		

Document Change History

This is the first release of the *intelligent Workload Distribution 8.0 WebSphere MQ Capture Adapter Reference Guide.* In the future, this section will list topics that are new or that have changed significantly since the first release of this document.

Chapter

1 iWD WebSphere MQ Capture Adapter Overview

This chapter explains the iWD WebSphere MQ Capture Adapter architecture and the components that are involved. The information in this chapter is organized into the following topics:

What is the iWD WebSphere MQ Capture Adapter?

The iWD WebSphere MQ Capture Adapter provides a way to connect third-party task-originating systems (such as BPM or workflow solutions) to iWD by using the IBM WebSphere MQ interface. The WebSphere MQ Capture Adapter provides a fully bidirectional link and supports the full iWD API (such as task creation, updating, holding and canceling as well as various task state change notifications).

iWD WebSphere MQ Capture Adapter Architecture

The iWD WebSphere MQ Capture Adapter uses the WebSphere MQ API to connect to a WebSphere MQ system. In iWD configuration, a WebSphere MQ capture point, which is a specific configuration object, is set up. The WebSphere MQ Capture Adapter is a technology solution that provides the actual integration point to the IBM WebSphere MQ system, whereas the

capture point is a configuration object against which business rules will be defined. Figure 1 illustrates the architecture.

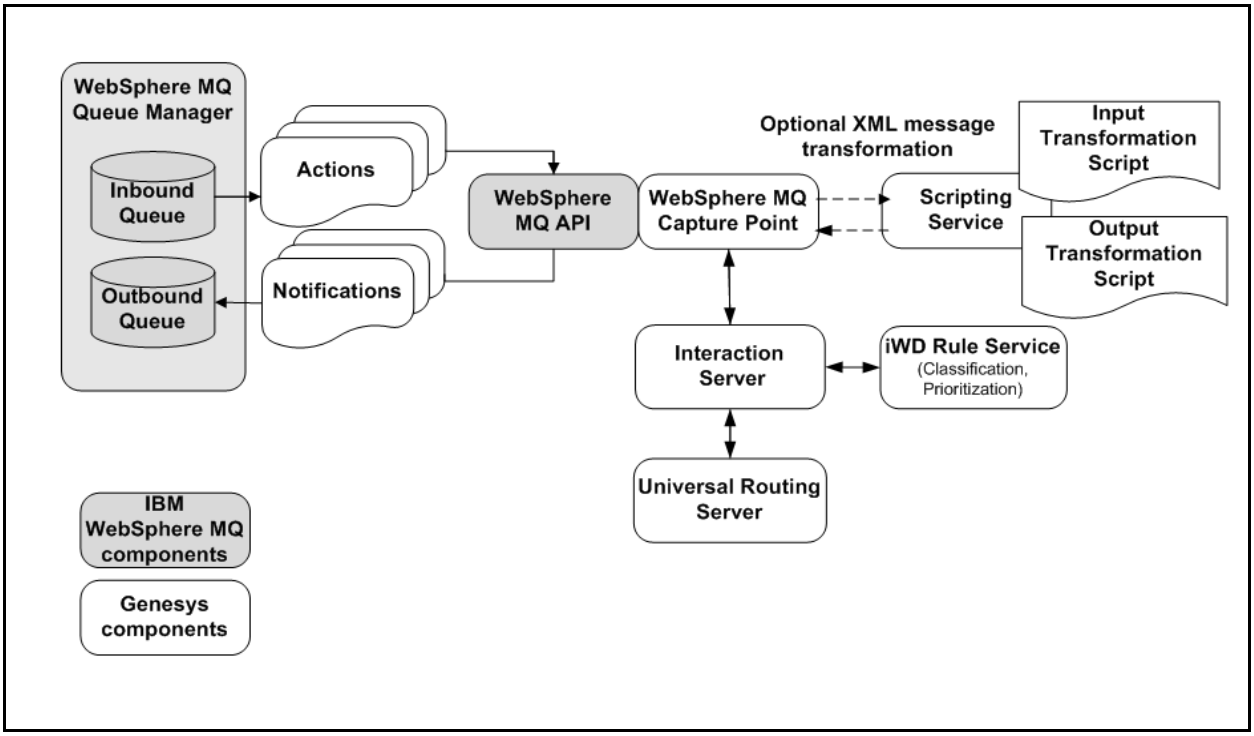

Figure 1: WebSphere MQ Capture Adapter Architecture

The WebSphere MQ Capture Adapter connects to a configurable MQ Queue Manager and utilizes two queues: input (`Inbound`) and output (`Outbound`). The respective names of the queues are configurable by using the `queueInbound` and `queueOutbound` properties (described in Table 1 on page 19). Prior to 8.0, these queues were called `GTLIn` and `GTLOut`, respectively.

- The input queue is used to receive messages from the task-originating system, such as task creation or task-update requests.
- The output queue is used to send back responses to requests, as well as to send notifications about task-state changes that are not triggered by the originating system (such as when a task is assigned to an agent).

Note: The WebSphere MQ Capture Adapter, like all others in iWD, supports multiple capture points.

The WebSphere MQ Capture Adapter supports optional message transformation. In this case, each incoming XML message, as well as each outgoing message, is passed through transformation scripts—thus, allowing integration with custom XML formats. Figure 2 depicts the transformation process.

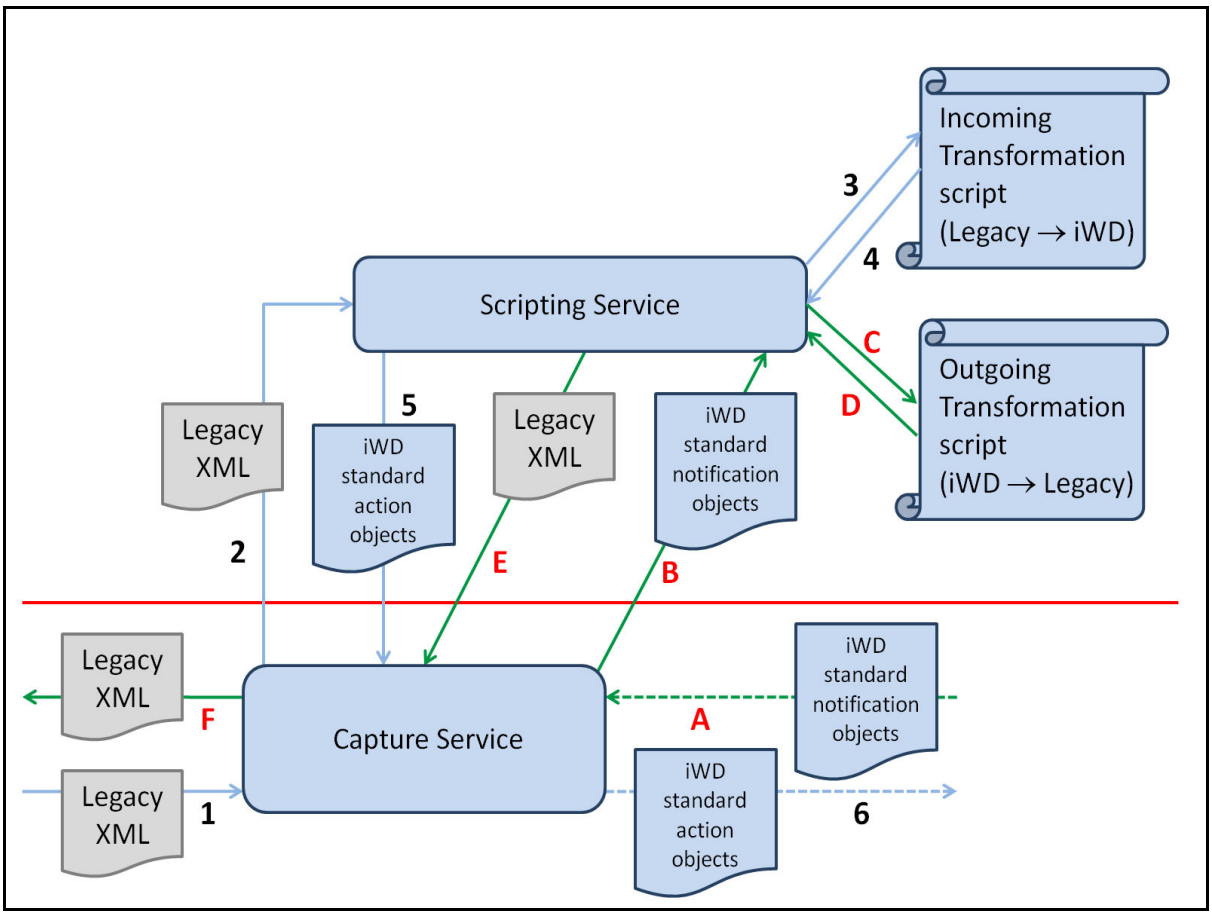

Figure 2: Transformation

The numbers in Figure 2 identify the steps in the flow when transformation is used:

1. The Legacy XML is captured.
2. The Legacy XML is passed to the Scripting Service.
3. The Scripting Service executes the Incoming Transformation script, using the Legacy XML file as input.
4. The Incoming Transformation script produces iWD standard action objects as output, which are passed to the Scripting Service.
5. The Scripting Service passes the iWD standard action objects to the Capture Service.
6. The task is created. (The dotted lines indicate communication with other iWD components).

The letters identify the steps in the flow for the reverse process:

A. The Capture Service receives notification about a task.

B. The Capture Service gets task notification objects and passes them to the Scripting Service.

C. The Scripting Service executes the Outgoing Transformation script, using the iWD standard notification objects as input.

D. The Outgoing Transformation script produces the Legacy XML as output and passes it to the Scripting Service.

E. The Scripting Service passes the newly created Legacy-format XML to the Capture Service.

F. The Capture Service writes it to an MQ or XML file to be used to respond back to the legacy system about requests or task state change notifications (for example, if the task has been assigned to an agent).

If transformation is not used, the steps that appear above the red line in Figure 2 are not invoked. In this case, the incoming messages from the legacy system must be in iWD standard XML format. This document describes the standard iWD XML message formats that are supported without the transformation. Each message is formatted as an XML string in which the root node is always named `GTLMessages`. The child nodes of `GTLMessages` indicate the message type, such as `CreateTask`.

The types and formats of the input and output messages that are supported are described in Chapter 2, "iWD MQ Messages," on page 23.

iWD WebSphere MQ Capture Point Installation

The WebSphere MQ Capture Point Service is an iWD service for capturing and manipulating tasks via the WebSphere Message Queue interface. The messages must be in XML format.

iWD provides a standard iWD XML message schema (described in detail in Chapter 2, "iWD MQ Messages," on page 23). Alternatively, it is possible to use custom XML formats via message-transformation scripts.

To handle custom XML formats, transformation scripts must be created for each direction that is needed: typically one for input transformation, and one for output. iWD currently supports the Groovy scripting language, as well as JavaScript, for these purposes. For more information on transformation scripts, see the standard iWD input and output transformation scripts that are included in the iWD core package.

This section describes how to deploy the WebSphere MQ Capture Point Adapter and configure its corresponding service.

Task Summary: Deploying the WebSphere MQ Capture Adapter

Objective	Related Procedures and Actions
1. Deploy the WebSphere MQ Capture Adapter	Install the WebSphere MQ Capture Adapter. Procedure: Installing the WebSphere MQ Capture Adapter, on page 16..
2. Create the WebSphere MQ Capture Point Service	Import the service template and configure the service. 1. Procedure: Importing the WebSphere MQ Capture Point Service template, on page 17. 2. Procedure: Configuring the WebSphere MQ Capture Service, on page 18.

Deploying the WebSphere MQ Capture Adapter

Before installing the adapter, you need to ensure your environment meets the following requirements and prerequisites:

- iWD 8.0 has been installed and configured. Refer to the *iWD 8.0 Deployment Guide* for more information.

Note: The WebSphere MQ Capture Adapter is supported on the same platforms as the rest of the iWD 8.0 components.

Procedure: Installing the WebSphere MQ Capture Adapter

Purpose: To install an instance of the WebSphere MQ Capture Adapter.

Note: iWD components, including the WebSphere MQ Capture Adapter, need to have their installation packages run two times: once specifying the iWD Manager instance, and once specifying the iWD Runtime Node instance. In other words, if the installation package is installed once, it is not actually installed and ready to be used. iWD Runtime Node and iWD Manager provide basic funcionality, and the remaining components deliver extended functionality. Due to the structure of web applications, they need to be installed "into" the basic applications (iWD Manager and iWD Runtime Node).This is outlined in the following procedure.

Start of procedure

1. From the server that is running iWD Manager, locate and double-click `Setup.exe` on the iWD WebSphere MQ Capture Adapter CD.
2. Click `Next` on the `Welcome` screen.
3. iWD components need to be installed into iWD Manager as well as into iWD Runtime Node. At this time, select the iWD Manager option to install iWD Rules into the iWD Manager instance and click `Next`.
4. Select the appropriate iWD Manager instance from the list that is displayed and click `Next`.
5. Choose a destination folder for iWD WebSphere MQ Capture Adapter. Either accept the default location or browse to a different location. Click `Next`.
6. Click `Install` to install the WebSphere MQ Capture Adapter into iWD Manager. Click `Finish` when the installation is complete.
7. From the server that is running iWD Runtime Node (which may or may not be the same server as the one running iWD Manager), again locate and double-click `Setup.exe` on the iWD WebSphere MQ Capture Adapter CD.
8. Click `Next` on the `Welcome` screen.
9. If you are installing on the same host as for iWD Manager, you will be presented with a `Maintenance Setup Type` screen. Select `Install a new Instance of the application` and click `Next`.
10. This time, select the iWD Runtime Node option for installation and click `Next`.
11. Select the appropriate iWD Runtime Node instance from the list that is displayed and click `Next`.

12. Choose a destination folder for iWD WebSphere MQ Capture Adapter. Either accept the default location or browse to a different location. Click `Next`.
13. Click `Install` to instal liWD WebSphere MQ Capture Adapter into iWD Runtime Node. Click `Finish` when the installation is complete.

End of procedure

Next Steps

- Procedure: Importing the WebSphere MQ Capture Point Service template.

Creating the WebSphere MQ Capture Point Service

The WebSphere MQ Capture Point Service template is not imported into the iWD configuration database by default. This section describes how to import the service template and configure the service.

Procedure: Importing the WebSphere MQ Capture Point Service template

Purpose: To import the WebSphere MQ Capture Point Service template and assign it as a Module for your Solution.

Start of procedure

1. Log into iWD Manager.

> **Note:** For a detailed description of the iWD Manager interface, including logging in, the interface layout, and available functionality, refer to the *iWD 8.0 Deployment Guide* and the *iWD Manager Help*.

2. Select the System tenant from the tenant selection drop-down list.
3. In the navigation tree, select `Import/Export`.
4. In the Import/Export details view, click `Browse` and navigate to the `config` sub-directory of the installation directory specified in Step 5 of Procedure: Installing the WebSphere MQ Capture Adapter, on page 16.
5. Select the `iwd_wmq_capture.xml` file to import.
6. Click `Import`. A message will appear to indicate that the import was successful.
7. Select your tenant from the list of managed tenants.

8. Move the iWD WebSphere MQ Capture Service to the Assigned Modules list. Click `Save`.
9. Select the tenant from the tenant selection drop-down list.
10. Select the `Services` navigation section, and then select your Solution in the navigation tree.
11. In the Solution Details view, move the iWD WebSphere MQ Capture Service to the Assigned Modules list. Click `Save`.
12. The iWD WebSphere MQ Capture Service is now available under the Service templates for your solution.

End of procedure

Next Steps

- Configure the WebSphere MQ Capture Service. See Procedure: Configuring the WebSphere MQ Capture Service.

Procedure: Configuring the WebSphere MQ Capture Service

Purpose: To create the service in iWD Manager based on the template, and to configure the properties for the service.

Start of procedure

1. In iWD Manager, select the Services navigation section.
2. Locate your solution in the navigation tree. Expand the Services node in the navigation tree (if necessary), and click `New Service`.
3. Select the iWD WebSphere MQ Capture Service from the templates drop-down list, and configure the remaining properties as described in Table 1 on page 19.
4. On the Service details screen, ensure that the correct iWD Runtime Node is selected if your environment has multiple runtime nodes. This ensures a more scalable solution by allowing you to distribute the iWD services across multiple nodes, and is also used for high availability configurations in which you would want primary and backup WebSphere MQ Capture Point services to be running on separate runtime nodes.
5. When configuration is complete, click `Save`. Remember that all changes have to be deployed before they can take effect.

End of procedure

Table 1 lists the configurable properties for the WebSphere MQ Capture Point Service.

Table 1: WebSphere MQ Capture Point Service Properties

Property	Description
startAutomatically	Indicates whether the service should be started automatically after the configuration deployment.
logLevel	The Service log level. This should be set to `Default` unless you are instructed otherwise by Genesys Technical Support. The log levels are the following: • `Default`—The service will use the logging level configured in the Logging Service. • `Debug`—The most detailed informational events that are most useful in debugging an application. • `Info`—Informational messages that highlight the progress of the application. • `Warn`—Potentially harmful situations. • `Error`—Error events that might not affect the application's ability to run. • `Fatal`—Severe error events. • `All`—Turns on all logging. • `Off`—Turns off all logging.
checkIfAlreadyCaptured	If set to `true`, iWD will verify whether another task that has a given `captureId` has already been captured.
timezone	The time zone of the WebSphere MQ capture point. Date/ time values will be converted from the specified time zone to UTC, before those values are stored in iWD. Also, any date/time values that are included in response messages will be converted to the specified time zone. If this parameter is not specified, it defaults to the iWDtenant time zone.
defaultTaskExpirationInDays	The default task expiration date. All tasks that are captured by this capture point will expire after the specified amount of days. This value can be overridden by explicitly specifying the task `expirationDateTime` attribute in the capture data.
stopInteractionOnComplete	If selected, when `completeTask` is issued by the source system, the interaction will be stopped in Interaction Server automatically.
interactionServerConnector	Select the Interaction Server Connector service. **Note:** The Interaction Server Connector must be configured before the WebSphere MQ Capture Point is configured.

Table 1: WebSphere MQ Capture Point Service Properties (Continued)

Property	Description
mqHost	The WebSphere MQ server address.
mqPort	The WebSphere MQ server port.
userName	The WebSphere MQ server username.
password	The WebSphere MQ server password.
queueManagerName	The WebSphere MQ queue manager name.
managerChannel	The WebSphere MQ queue manager channel.
managerCCSID	The WebSphere MQ queue manager Coded Character Set Identifier (CCSID).
messageEncoding	The WebSphere MQ message encoding.
queueInbound	The inbound queue from which the iWD messages will be retrieved.
queueOutbound	The outbound queue to which the iWD notifications will be submitted.
queueWorkerThreads	Performance tuning: The size of the thread pool. It is recommended to set to this property to 1 if the order of incoming message processing is important.
reconnectTimeoutSeconds	Exception handling*:* The connection timeout.
backupFor	High availability: The primary WebSphere MQ capture point in a high availability setup scenario. The service will perform a backup role in such a setup, and it will take over processing if the primary service becomes unavailable. For more information about high availability, refer to the *iWD 8.0 Deployment Guide.*
TransformScriptingService	Optional dependency: Scripting Service (see the *iWD 8.0 Deployment Guide* for more information about the Scripting Service). If specified, XML input/output will be transformed by using input/output transformation scripts.
TransformInputScript	The input transformation script; must be specified when `TransformScriptingService` is selected.

Table 1: WebSphere MQ Capture Point Service Properties (Continued)

Property	Description
TransformOutputScript	The output transformation script; must be specified when `TransformScriptingService` is selected.
TransformOutputRootNode	The name of the XML root node in output files. This property is visible only when `TransformScriptingService` is selected.
NotifyError	If selected, indicates that error notifications will be sent to an outbound queue; available only when transformation is not enabled.
NotifyTaskAssigned	If selected, indicates that “task assigned” notifications will be submitted to an outbound queue; available only when transformation is not enabled.
NotifyTaskCanceled	If selected, indicates that “task canceled” notifications will be submitted to an outbound queue; available only when transformation is not enabled.
NotifyTaskCreated	If selected, indicates that “task created” notifications will be submitted to an outbound queue; available only when transformation is not enabled.
NotifyTaskRejected	If selected, indicates that “task rejected” notifications will be submitted to an outbound queue; available only when transformation is not enabled.
NotifyTaskCompleted	If selected, indicates that “task completed” notifications will be submitted to an outbound queue; available only when transformation is not enabled.
NotifyTaskDistributedQueue	If selected, indicates that “task moved into distribution queue” notifications will be submitted to an outbound queue; available only when transformation is not enabled.
NotifyTaskFinished	If selected, indicates that “task finished” notifications will be submitted to an outbound queue; available only when transformation is not enabled.
NotifyTaskHeld	If selected, indicates that “task held” notifications will be submitted to an outbound queue; available only when transformation is not enabled.
NotifyTaskRestarted	If selected, indicates that “task restarted” notifications will be submitted to an outbound queue; available only when transformation is not enabled.

Table 1: WebSphere MQ Capture Point Service Properties (Continued)

Property	Description
NotifyTaskResumed	If selected, indicates that “task resumed” notifications will be submitted to an outbound queue; available only when transformation is not enabled.
NotifyTaskUpdated	If selected, indicates that “task updated” notifications will be submitted to an outbound queue; available only when transformation is not enabled.

Chapter

2 iWD MQ Messages

This chapter provides a detailed description of all of the input and output iWD MQ Messages that are supported by the capture adapter.

The information in this chapter is organized in the following sections:

Data Types

Table 2 describes the data types that are used in iWD MQ messages.

Table 2: Data Types for iWD MQ Messages

Type	Description
Integer	An integer value (-2^{31} <value <2^{31}).
String	A string value. The maximum length is specified in parentheses, where applicable.
Boolean	A Boolean value (`true` or `false`).
DateTime	A date/time value. Date/time should be formatted according to the ISO 8601 standard `YYYY-MM-DDThh:mm:ss` (for example 2007-08-26T21:32:00) and should be provided for the time zone that is configured for the given MQ Capture Point.

Messages

The following information is documented for each message:

- Direction—"In" or "Out".Every "In" message comes from the originating system and results in an "Out" message, unless the specific "Out" message is turned off in capture point configuration. "Out" messages are sent as responses to "In" messages, as well as notifications when the task state changes within iWD.
- Description—A functional description of the message.
- Format—The XML format of the message. This illustrates a structure of the entire XML message, by using data types (see Table 2 on page 23) instead of node values.
- Attributes—A description of each attribute that is used in the XML message.
- Response messages—The response messages that this message can trigger. Response messages are applicable only for messages that have an "In" direction.
- Error codes—The error codes that this message can return by using the "Error" message. Error codes are applicable only for messages that have an "In" direction.

Task Action

Direction: In

Description: This section describes common attributes and responses for all inbound messages.

Format:

```
<GTLMessages>
   <ActionName>
      <InteractionId>Integer</InteractionId> or
<CaptureId>String(64)</CaptureId>
      <Actor>String(255)</Actor>
      <Reason>String(255)</Reason>
      <ActionDateTime>DateTime</ActionDateTime>
      [attributes specific to action]
   </ActionName>
</GTLMessages>
```

Attributes: See Table 3.

Table 3: Attributes for Task Action Messages

Attribute	Description
InteractionId	The task's interaction ID. This is a unique ID assigned by Interaction Server.
CaptureId	The task's ID in the originating system.
Actor (optional)	The user or system that triggered the message. This is a free-form text field that is used for auditing purposes and will be set to name of the capture point, if none is provided.
Reason (optional)	The reason that the message was submitted. This is a free-form text field that is used for auditing purposes.
ActionDateTime (optional)	The date/time when the action was triggered. This will be set to the current date/time that the message is processed, if none is provided.

Response messages: Action-specific response messages or error messages (see "Error" on page 28).
Error codes: See Table 4.

Table 4: Error Codes for Task Action Messages

Error code	Description
INVALID_FORMAT	The message is not formatted correctly.
TASK_NOT_FOUND	The task that has the requested CaptureId or InteractionId is not found. This error code can be triggered for all action messages, except for the CreateTask message (see "CreateTask" on page 26).

Task Notification

Direction: Out
Description: This section describes common attributes and responses for all outbound messages.
Format:

```
<GTLMessages>
   <[notification name]>
      <InteractionId>Integer</InteractionId>
      <CaptureId>String(64)</CaptureId>
      <CapturePointId>String(16)</CapturePointId>
      <DistributionPointId>String(16)</DistributionPointId>
      <Actor>String(255)</Actor>
```

```
      <Reason>String(255)</Reason>
      <EventDateTime>DateTime</EventDateTime>
      [attributes specific to notification]
   </[notification name]>
</GTLMessages>
```

Attributes: See Table 5.

Table 5: Attributes for Task Notification Messages

Attribute	Description
InteractionId	The task's interaction ID. This is a unique ID assigned to the task by Interaction Server.
CaptureId	The task's ID in the originating system.
CapturePointId	The service ID of the capture point by which the task was captured.
DistributionPointId	This attribute is provided for backward compatibility.
Actor	The user or system that triggered the notification.
Reason	The reason for the notification.
EventDateTime	The date/time when the notification was triggered.

CreateTask

Direction: In

Description: Creates a new task in iWD and populates it with the provided attributes. All attributes of this message are optional. Typically, most of the core task attributes, such as `ProcessId`, `Priority`, and `BusinessValue`, are calculated and assigned with the iWD rules and, therefore, should be left out. Interaction Server assigns a unique Interaction ID for each task. Interaction Server stores and maintains the IDs in the Interaction Server database.

Format:

```
<GTLMessages>
   <CreateTask>
```

Standard action attributes, as documented in "Task Action" on page 24, except for `InteractionId`.

```
      <channel>String(32)</channel>
      <category>String(32)</category>
      <activationDateTime>DateTime</activationDateTime>
      <dueDateTime> DateTime </dueDateTime>
      <expirationDateTime> DateTime </expirationDateTime>
      <businessValue> DateTime </businessValue>
      <priority>Integer</priority>
      <processId>String(16)</processId>
      <Ext>
         <customerID>String(64)</customerID>
```

```
            <customerSegment> String(64)</customerSegment>
            <productType> String(64)</productType>
            <productSubtype> String(64)</productSubtype>
            <resultCode> String(64)</resultCode>

<sourceFirstCreatedDateTime>DateTime</sourceFirstCreatedDateTime>
            <sourceCreatedDateTime>DateTime</sourceCreatedDateTime>
            <sourceDueDateTime>DateTime</sourceDueDateTime>
            <sourceProcessType> String(64)</sourceProcessType>
            <sourceProcessSubtype> String(64)</sourceProcessSubtype>
            <sourceTenant> String(64)</sourceTenant>
         </Ext>
         <Data>
            <CustomAttribute1> String(255)</CustomAttribute1>
            ...
         </Data>
         <Hold>Boolean</Hold>
      </CreateTask>
</GTLMessages>
```

Attributes: See Table 6.

Table 6: Attributes for CreateTask Messages

Attribute	Description
Hold	Whether to hold the task initially. If `true`, the task will be created with its initial status set to `NewHeld` and will not be processed further, until a subsequent `ResumeTask` message (see "ResumeTask" on page 38).
CaptureId (optional)	If a `CaptureId` is not provided, it will be assigned to the same generated value as `InteractionId`.

See "Task Action" on page 24 and "TaskInfo" on page 29 for the description of the remaining attributes.

Response message: `TaskCreated` (see "TaskCreated" on page 28).

Error code: see Table 7.

Table 7: Error Codes for CreateTask Messages

Error code	Description
TASK_ALREADY_CAPTURED	If the captures point's `checkIfAlreadyCaptured` flag is enabled, iWD will check whether a task that has a given `captureId` already exists in the database. If this is the case, the task will not be captured, and an error message that has the `TASK_ALREADY_CAPTURED` code will be submitted to the `Outbound` queue.

TaskCreated

Direction: Out
Description: The `TaskCreated` message is submitted as a response to the `CreateTask` message (see page 26) and indicates successful task creation.
Format:

```
<GTLMessages>
   <TaskCreated>
      Standard notification attributes, as documented in "Task Notification"
on page 25.
   </TaskCreated>
</GTLMessages>
```

Error

Direction: Out
Description: The `Error` message is submitted as a response to iWD request messages, indicating that the requested operation has failed.
Format:

```
<GTLMessages>
   <Error>
      <Message>String</Message>
      <Code>String</Code>
      <Parameter>String</Parameter>
      <Parameter>String</Parameter>
      ...
   </Error>
</GTLMessages>
```

Attributes: See Table 8.

Table 8: Attributes for Error Messages

Attribute	Description
Message	The formatted error message.
Code	The error code (string).
Parameter	The error parameter. There can be zero, one, or multiple error parameters. The number of parameters is specific to each error code.

GetTaskInfo

Direction: In
Description: Requests task details by using the given task's capture ID or interaction ID.
Format:

```
<GTLMessages>
   <GetTaskInfo>
      Standard action attributes, as documented in "Task Action" on page 24.
   </GetTaskInfo>
</GTLMessages>
```

Response message: "TaskInfo".

TaskInfo

Direction: Out

Description: The `TaskInfo` message is submitted as a response to the `GetTaskInfo` message (see "GetTaskInfo" on page 28) and provides detailed information about the requested task.

Format:

```
<GTLMessages>
   <TaskInfo>
      Standard action attributes, as documented in "Task Action" on page 24.
      <tenantId>String(16)</tenantId>
      <solutionId> String(16)</solutionId>
      <DepartmentId> String(16)</DepartmentId>
      <processId> String(16)</processId>
      <channel>String(32)</channel>
      <category> String(32)</category>
      <status> String(16)</status>
      <businessCalendarId> String(16)</businessCalendarId>
      <createdDateTime>DateTime</createdDateTime>
      <heldDateTime>DateTime</heldDateTime>
      <assignedDateTime>DateTime</assignedDateTime>
      <completedDateTime>DateTime</completedDateTime>
      <activationDateTime>DateTime</activationDateTime>
      <dueDateTime>DateTime</dueDateTime>
      <expirationDateTime>DateTime</expirationDateTime>
      <priority>Integer</priority>
      <reprioritizeDateTime>DateTime</reprioritizeDateTime>
      <businessValue>Integer</businessValue>
      <assignedToUser>String(64)</assignedToUser>
      <Queue>String(255)</Queue>
      <QueueType>String(16)</QueueType>
      <QueueTarget>String(255)</QueueTarget>
      <Ext>
         <customerID>String(64)</customerID>
         <customerSegment> String(64)</customerSegment>
         <productType> String(64)</productType>
         <productSubtype> String(64)</productSubtype>
         <resultCode> String(64)</resultCode>

<sourceFirstCreatedDateTime>DateTime</sourceFirstCreatedDateTime>
         <sourceCreatedDateTime>DateTime</sourceCreatedDateTime>
         <sourceDueDateTime>DateTime</sourceDueDateTime>
         <sourceProcessType> String(64)</sourceProcessType>
         <sourceProcessSubtype> String(64)</sourceProcessSubtype>
```

```
      <sourceTenant> String(64)</sourceTenant>
    </Ext>
    <Data>
      <customAttribute1> String(255)</customAttribute1>
      ...
    </Data>
  <TaskInfo>
<GTLMessages>
```

Attributes: See Table 9.

Table 9: Attributes for TaskInfo Messages

Attribute	Description
tenantId	The task's tenant ID, as configured in iWD Manager, assigned as soon as the task is created. This attribute is submitted to the CIM Platform with the `IWD_tenantId` key; updates within the CIM Platform are ignored.
solutionId	The tasks's solution instance ID, as configured in iWD Manager, assigned as soon as the task is created. This attribute is submitted to the CIM Platform with the `IWD_solutionId` key (even if it is excluded by a filter); updates within the CIM Platform are ignored.
DepartmentId	The tasks's department ID, as configured in iWD Manager, assigned when the task's process is identified either by iWD rules or explicitly by the task-originating system. This attribute is submitted to the CIM Platform with the `IWD_DepartmentId` key; updates within the CIM Platform are ignored.
processId	The tasks's process ID, as configured in iWD Manager, assigned when the task's process is identified either by iWD rules or explicitly by the task-originating system. This attribute is submitted to the CIM Platform with the `IWD_processId` key; updates within the CIM Platform are ignored.
channel	The task's media channel—for example Fax, E-mail, or Webform. This attribute is submitted to the CIM Platform with the `IWD_channel` key; updates within the CIM Platform are picked up.
category	The task's category—for example `Followup`. This attribute is submitted to the CIM Platform with the `IWD_category` key; updates within the CIM Platform are picked up.

Table 9: Attributes for TaskInfo Messages (Continued)

Attribute	Description
status	Task status: `New`—The task has just been created and will be processed. `Captured`—The task has been processed, but it is not yet prioritized. `Queued`—The task is processed and prioritized at least once. `Assigned`—The task is assigned to an agent. `Completed`—The task is completed. `Held`—The task is held and will not be reprioritized or distributed until it is resumed. `Error`—An error has occurred during task processing, prioritization, or distribution. Error details are stored in the `Error` custom, extended task attribute. The task can be restarted, and the iWD will attempt to process the task again. `Canceled`—The task is canceled. `Rejected`—The task has been rejected during processing. This can occur when the task is assigned to an expired department or process.
businessCalendarId	The ID of the business calendar that is assigned to the task, as configured in iWD Manager. **Note:** Maximum length is 16 characters
createdDateTime	The date/time when the task has been created in iWD. This attribute is submitted to the CIM Platform with the `IWD_dueDateTime` key; updates within the CIM Platform are ignored.
heldDateTime	The date/time when the task has been held (set only when task status is either `Held` or `Error`).
assignedDateTime	The date/time when the task has been assigned.
completedDateTime	The date/time when the task has been completed.
activationDateTime	The date and time when the task becomes active; before that, it will stay queued and will not be reprioritized and distributed. If this is not set, the task becomes active instantly.
dueDateTime	The date and time by which the task should be completed, according to the service-level agreement (SLA). This attribute is submitted to the CIM Platform with the `IWD_dueDateTime` key; updates within the CIM Platform are picked up.
expirationDateTime	The date and time when the task expires and will be archived. Only tasks that have been `Canceled`, `Completed`, or `Rejected` are archived.

Table 9: Attributes for TaskInfo Messages (Continued)

Attribute	Description
priority	The task priority, which is an integer value that is used to order tasks. The higher the value, the higher that the task will stand in the queue and the sooner that it will be routed. This attribute is submitted to the CIM Platform with the `Priority` key; updates in the CIM Platform are picked up.
reprioritizeDateTime	The date/time when the task should be reprioritized; if this is set to `null`, no more reprioritization will be done. This value is normally updated during prioritization, based on rule expressions, such as "Reprioritize in 5 minutes".
businessValue	The business value of the task. This attribute is submitted to the CIM Platform with the `IWD_businessValue` key; updates in the CIM Platform are picked up.
assignedToUser	The user ID to which a task is assigned, as supplied by the CIM Platform.
Queue	The distribution's queue name.

QueueType	The type of distribution queue:
	`InteractionQueue`
	`AgentWorkbin`
	`AgentGroupWorkbin`
	`PlaceWorkbin`
	`PlaceGroupWorkbin`

QueueTarget	The queue target—for example, `Agent ID`, if the queue type is `AgentWorkbin`.
customerId	The customer's ID. This attribute is submitted to the CIM Platform with the `IWD_ext_customerId` key; updates in the CIM Platform are picked up.
customerSegment	The customer's segment or value. This attribute is submitted to the CIM Platform with the `IWD_ext_customerSegment` key; updates in the CIM Platform are picked up.
productType	The related product—for example, DSL. This attribute is submitted to the CIM Platform with the `IWD_ext_productType` key; updates in the CIM Platform are picked up.
productSubtype	The subtype of the related product—for example, PremiumDSL. This attribute is submitted to the CIM Platform with the `IWD_ext_productSubtype` key; updates in the CIM Platform are picked up.

Table 9: Attributes for TaskInfo Messages (Continued)

Attribute	Description
resultCode	The task result code/outcome; typically, it is set by an agent in a softphone or another client application. This attribute is submitted to the CIM Platform with the `IWD_ext_resultCode` key; updates in the CIM Platform are picked up.
sourceFirstCreatedDateTime	The earliest timestamp of the task in the enterprise; it is applicable if there is another system, such as a fax server, that is used before the task originating system. This attribute is submitted to the CIM Platform with the `IWD_ext_sourceFirstCreatedDateTimeTime` key; updates in the CIM Platform are ignored.
sourceCreatedDateTime	The task-creation timestamp in the task-originating system. This attribute is submitted to the CIM Platform with the `IWD_ext_sourceCreatedDateTime` key; updates in the CIM Platform are ignored.
sourceDueDateTime	The task-due timestamp in the task-originating system. This attribute is submitted to the CIM Platform with the `IWD_ext_sourceDueDateTime` key; updates in the CIM Platform are ignored.
sourceProcessType	A related process in the task-originating system—for example: Order. This attribute is submitted to the CIM Platform with the `IWD_ext_sourceProcessType` key; updates in the CIM Platform are ignored.
sourceProcessTypeSubtype	The subtype of the related process in the task-originating system. This attribute is submitted to the CIM Platform with the `IWD_ext_sourceProcessSubtype` key; updates in the CIM Platform are ignored.
sourceTenant	The tenant ID or name in the task-originating system. This attribute is submitted to the CIM Platform with the `IWD_ext_sourceTenant` key; updates in the CIM Platform are ignored.
data	Custom task attributes. These attributes can be used to associate additional task originating system-specific data to the task that can be used in iWD rules, routing, and historical reporting.

UpdateTask

Direction: In

Description: Updates the attributes of the task that has the given task's capture ID or interaction ID. This results in the interaction properties of the task being updated through the Genesys Interaction Server.

All attributes except for `CaptureId` and `InteractionId` are optional. If the attribute is not provided, it will not be updated.

Format:

```
<GTLMessages>
   <UpdateTask>
```

Standard notification attributes, as documented in "Task Action" on page 24.

```
      <category>String(32)</category>
      <activationDateTime>DateTime</activationDateTime>
      <dueDateTime> DateTime </dueDateTime>
      <expirationDateTime> DateTime </expirationDateTime>
      <businessValue>Integer</businessValue>
      <priority>Integer</priority>
      <ext>
         <customerID>String(64)</customerID>
         <customerSegment> String(64)</customerSegment>
         <productType> String(64)</productType>
         <productSubtype> String(64)</productSubtype>
         <resultCode> String(64)</resultCode>

<sourceFirstCreatedDateTime>DateTime</sourceFirstCreatedDateTime>
         <sourceCreatedDateTime>DateTime</sourceCreatedDateTime>
         <sourceDueDateTime>DateTime</sourceDueDateTime>
         <sourceProcessType> String(64)</sourceProcessType>
         <sourceProcessSubtype> String(64)</sourceProcessSubtype>
         <sourceTenant> String(64)</sourceTenant>
      </ext>
      <data>
         <customAttribute1> String(255)</customAttribute1>
         ...
      </data>
   </UpdateTask>
</GTLMessages>
```

Attributes: See "Task Action" on page 24 and "TaskInfo" on page 29 for a description of the attributes.

Response message: "TaskUpdated".

TaskUpdated

Direction: Out

Description: The `TaskUpdated` message is submitted as a response to the `UpdateTask` message (see "UpdateTask" on page 33), as well as when the task is updated either via the iWD Manager or within the CIM Platform.

Format:

```
<GTLMessages>
   <TaskUpdated>
```

Standard notification attributes, as documented in"Task Notification" on page 25.

```
      <tenantId>String(16)</tenantId>
      <solutionId> String(16)</solutionId>
```

```
      <DepartmentId> String(16)</DepartmentId>
      <processId> String(16)</processId>
      <channel>String(32)</channel>
      <category> String(32)</category>
      <status> String(16)</status>
      <businessCalendarId> String(16)</businessCalendarId>
      <createdDateTime>DateTime</createdDateTime>
      <heldDateTime>DateTime</heldDateTime>
      <assignedDateTime>DateTime</assignedDateTime>
      <completedDateTime>DateTime</completedDateTime>
      <activationDateTime>DateTime</activationDateTime>
      <dueDateTime>DateTime</dueDateTime>
      <expirationDateTime>DateTime</expirationDateTime>
      <priority>Integer</priority>
      <reprioritizeDateTime>DateTime</reprioritizeDateTime>
      <businessValue>Integer</businessValue>
      <assignedToUser>String(64)</assignedToUser>
      <ext>
         <customerID>String(64)</customerID>
         <customerSegment> String(64)</customerSegment>
         <productType> String(64)</productType>
         <productSubtype> String(64)</productSubtype>
         <resultCode> String(64)</resultCode>

<sourceFirstCreatedDateTime>DateTime</sourceFirstCreatedDateTime>
         <sourceCreatedDateTime>DateTime</sourceCreatedDateTime>
         <sourceDueDateTime>DateTime</sourceDueDateTime>
         <sourceProcessType> String(64)</sourceProcessType>
         <sourceProcessSubtype> String(64)</sourceProcessSubtype>
         <sourceTenant> String(64)</sourceTenant>
      </ext>
      <data>
         <customAttribute1> String(255)</customAttribute1>
         ...
      </data>
      <actor>String(255)</actor>
   <TaskUpdated>
<GTLMessages>
```

Attributes: See "Task Action" on page 24 and "TaskInfo" on page 29 for a description of the attributes.

TaskDistributedQueue

Direction: Out

Description: The `TaskDistributedQueue` message is submitted when the task is moved by the CIM Platform into any interaction queue or workbin, other than Interaction Server's predefined queues and workbins that are reserved for iWD.

Format:

```
<GTLMessages>
```

```
<TaskDistributedQueue>
```
Standard notification attributes, as documented in “Task Notification” on page 25.
```
<Queue>String(255)</Queue>
<QueueType>String(16)</QueueType>
<QueueTarget>String(255)</QueueTarget>
</TaskDistributedQueue>
</GTLMessages>
```
Attributes: See “Task Notification” on page 25 and “TaskInfo” on page 29 for a description of the attributes.

TaskAssigned

Direction: Out

Description: The `TaskAssigned` message is submitted when the task is assigned to an agent.

Format:
```
<GTLMessages>
<TaskAssigned>
```
Standard notification attributes, as documented in “Task Notification” on page 25.
```
<AssignedToUser>String(64)</AssignedToUser>
</TaskAssigned>
</GTLMessages>
```
Attributes: See “Task Notification” on page 25 and “TaskInfo” on page 29 for a description of the attributes.

CompleteTask

Direction: In

Description: Completes the task that has a given capture ID or interaction ID.

Format:
```
<GTLMessages>
<CompleteTask>
```
Standard action attributes, as documented in“Task Action” on page 24.
```
</CompleteTask>
</GTLMessages>
```
Attributes: See “Task Action” on page 24 for a description of the attributes.

Response message: “TaskCompleted”.

Error codes: See Table 10.

Table 10: Error Codes for CompleteTask Messages

Error Code	Description
CANNOT_COMPLETE_TASK	Cannot complete the task, because it is already completed, canceled, or rejected.

TaskCompleted

Direction: Out
Description: The `TaskCompleted` message is submitted as a response to the `CompleteTask` message (see "CompleteTask" on page 36), as well as when the task is placed into the predefined `CompletedQueue` interaction queue in Interaction Server.
Format:

```
<GTLMessages>
   <TaskCompleted>
      Standard notification attributes, as documented in "Task Notification" on page 25
   </TaskCompleted>
</GTLMessages>
```

Attributes: See "Task Notification" on page 25 for a description of the attributes.

HoldTask

Direction: In
Description: Holds the task that has given task's capture ID or interaction ID. As soon as it is held, the task will not be reprioritized or, potentially, assigned until it is resumed (see "ResumeTask" on page 38).
Only tasks that are not held, assigned, completed, canceled, or rejected can be held.
Format:

```
<GTLMessages>
   <HoldTask>
      Standard action attributes, as documented in "Task Action" on page 24.
   </HoldTask>
</GTLMessages>
```

Attributes: See "Task Action" on page 24 for a description of the attributes.
Response message: "TaskHeld".

Error codes: See Table 11.

Table 11: Error Codes for HoldTask Messages

Error code	Description
CANNOT_HOLD_TASK	Cannot hold the task because it is assigned, completed, canceled, rejected, or already held.

TaskHeld

Direction: Out
Description: The `TaskHeld` message is submitted as a response to the `HoldTask` message (see "HoldTask" on page 37), as well as when the task is held from the iWD Manager.
Format:

```
<GTLMessages>
   <TaskHeld>
      Standard notification attributes, as documented in "Task Notification" on page 25.
   </TaskHeld>
</GTLMessages>
```

Attributes: See "Task Notification" on page 25 for a description of the attributes.

TaskErrorHeld Message

Direction: Out
Description: The `TaskErrorHeld` message is submitted when the task is held because of a configuration error (such as incomplete rules).
Format:

```
<GTLMessages>
   <TaskErrorHeld>
      Standard notification attributes, as documented in "Task Notification" on page 25.
      <Error>String(255)</Error>
   </TaskErrorHeld>
</GTLMessages>
```

Attributes: See "Task Notification" on page 25 for a description of the attributes.

ResumeTask

Direction: In
Description: Resumes the held task that has the given task's capture ID or interaction ID.
As soon as it is resumed, the task will be processed and assigned normally, according to the iWD rules.
Only tasks that are held can be resumed.
Format:

```
<GTLMessages>
   <ResumeTask>
      Standard action attributes, as documented in "Task Action" on page 24.
   </ResumeTask>
</GTLMessages>
```

Attributes: See "Task Action" on page 24 for a description of the attributes.

Response message: "TaskResumed".
Error modes: See Table 12.

Table 12: Error Codes for ResumeTask Messages

Error code	Description
CANNOT_RESUME_TASK	Cannot resume the task, because it is not held.

TaskResumed

Direction: Out

Description: The `TaskResumed` message is submitted as a response to the `ResumeTask` message (see "ResumeTask" on page 38), as well as when a task is held from iWD Manager.

Format:

```
<GTLMessages>
   <TaskResumed>
      Standard notification attributes, as documented in "Task Notification" on page 25.
   </TaskResumed>
</GTLMessages>
```

Attributes: See "Task Notification" on page 25 for a description of the attributes.

RestartTask

Direction: In
Description: Restarts the task that has the given task's capture ID or interaction ID.
As soon as it is restarted, the task will be reclassified and reprioritized.
Only tasks that are not rejected can be restarted.
Format:

```
<GTLMessages>
   <RestartTask>
      Standard action attributes, as documented in "Task Action" on page 24.
   </RestartTask>
</GTLMessages>
```

Attributes: See "Task Action" on page 24 for a description of the attributes.
Response message: "TaskRestarted".
Error codes: See Table 13.

Table 13: Error Codes for RestartTask Messages

Error Code	Description
CANNOT_RESTART_TASK	Cannot restart the task because it is rejected.

TaskRestarted

Direction: Out
Description: The `TaskRestarted` message is submitted as a response to the `RestartTask` message (see "RestartTask"), as well as when the task is either restarted from the iWD Manager or moved to the predefined `New` interaction queue within the CIM Platform.
Format:

```
<GTLMessages>
   <TaskRestarted>
      Standard notification attributes, as documented in "Task Notification" on page 25.
   </TaskRestarted>
</GTLMessages>
```

Attributes: See "Task Notification" on page 25 for a description of the attributes.

CancelTask

Direction: In
Description: Cancels the task that has the given task's capture ID or interaction ID.
As soon as it is canceled, the interaction is moved to the `iWD_Canceled` queue. Tasks that have already been canceled cannot be canceled again. All other tasks can be canceled.
Format:

```
<GTLMessages>
   <CancelTask>
      Standard action attributes, as documented in "Task Action" on page 24.
   </CancelTask>
</GTLMessages>
```

Attributes: See "Task Action" on page 24 for a description of the attributes.
Response message: "TaskCanceled".
Error codes: See Table 14.

Table 14: Error Codes for CancelTask Messages

Error code	Description
CANNOT_CANCEL_TASK	Cannot cancel task, because it is canceled.

TaskCanceled

Direction: Out
Description: The `TaskCanceled` message is submitted as a response to the `CancelTask` message (see "CancelTask"), as well as when the task is canceled from iWD Manager.

Format:

```
<GTLMessages>
   <TaskCanceled>
      Standard notification attributes, as documented in "Task Notification" on page 25.
   </TaskCanceled>
</GTLMessages>
```

Attributes: See "Task Notification" on page 25 for a description of the attributes.

TaskRejected

Direction: Out

Description: The `TaskRejected` message is submitted when the task is rejected by the iWD Classification Service. The task can be rejected when a process or department to which the task is assigned is currently inactive (that is, either expired or not yet active).

Format:

```
<GTLMessages>
   <TaskRejected>
      Standard notification attributes, as documented in "Task Notification" on page 25.
   </TaskRejected>
</GTLMessages>
```

Attributes: See "Task Notification" on page 25 for a description of the attributes.

Ping

Direction: In

Description: A simple `Ping` message that can be used to check the health of the MQ Capture Point. The message includes an optional ID, which will be present in the corresponding `Pong` message.

Format:

```
<GTLMessages>
   <Ping>ID</Ping>
</GTLMessages>
```

Response message: "Pong".

Pong

Direction: Out

Description: Submitted as a response to the `Ping` message (see "Ping" on page 41), indicating that the MQ Capture Point service is active. The `Pong` message contains the ID that was sent in the `Ping` message.

Format:

```
<GTLMessages>
   <Pong>ID</Pong>
</GTLMessages>
```

Related Documentation Resources

The following resources provide additional information that is relevant to this software. Consult these additional resources as necessary.

intelligent Workload Distribution

- *iWD 8.0 Deployment Guide*, which provides information about how iWD is configured and managed. Specifically, the "Overview" chapter, which introduces all of the main iWD concepts.

Genesys

- *Genesys Technical Publications Glossary,* which ships on the Genesys Documentation Library DVD and which provides a comprehensive list of the Genesys and computer-telephony integration (CTI) terminology and acronyms used in this document.
- *Genesys Migration Guide*, which ships on the Genesys Documentation Library DVD, and which provides documented migration strategies for Genesys product releases. Contact Genesys Technical Support for more information.
- Release Notes and Product Advisories for this product, which are available on the Genesys Technical Support website at http://genesyslab.com/support.

Information about supported hardware and third-party software is available on the Genesys Technical Support website in the following documents:

- *Genesys Supported Operating Environment Reference Manual*
- *Genesys Supported Media Interfaces Reference Manual*

Consult these additional resources as necessary:

- *Genesys Hardware Sizing Guide,* which provides information about Genesys hardware sizing guidelines.

- *Genesys Interoperability Guide,* which provides information on the compatibility of Genesys products with various Configuration Layer Environments; Interoperability of Reporting Templates and Solutions; and Gplus Adapters Interoperability.
- *Genesys Licensing Guide,* which introduces you to the concepts, terminology, and procedures relevant to the Genesys licensing system.
- *Genesys Database Sizing Estimator 7.6 Worksheets,* which provides a range of expected database sizes for various Genesys products.

Document Conventions

This document uses certain stylistic and typographical conventions—introduced here—that serve as shorthands for particular kinds of information.

Document Version Number

A version number appears at the bottom of the inside front cover of this document. Version numbers change as new information is added to this document. Here is a sample version number:

80fr_ref_06-2008_v8.0.001.00

You will need this number when you are talking with Genesys Technical Support about this product.

Screen Captures Used in This Document

Screen captures from the product graphical user interface (GUI), as used in this document, may sometimes contain minor spelling, capitalization, or grammatical errors. The text accompanying and explaining the screen captures corrects such errors *except* when such a correction would prevent you from installing, configuring, or successfully using the product. For example, if the name of an option contains a usage error, the name would be presented exactly as it appears in the product GUI; the error would not be corrected in any accompanying text.

Type Styles

Table 15 describes and illustrates the type conventions that are used in this document.

Table 15: Type Styles

Type Style	Used For	Examples
Italic	• Document titles • Emphasis • Definitions of (or first references to) unfamiliar terms • Mathematical variables Also used to indicate placeholder text within code samples or commands, in the special case where angle brackets are a required part of the syntax (see the note about angle brackets on page 46).	Please consult the *Genesys Migration Guide* for more information. Do *not* use this value for this option. A *customary and usual* practice is one that is widely accepted and used within a particular industry or profession. The formula, $x + 1 = 7$ where x stands for . . .

Table 15: Type Styles (Continued)

Type Style	Used For	Examples
Monospace font (Looks like `teletype` or `typewriter text`)	All programming identifiers and GUI elements. This convention includes: • The *names* of directories, files, folders, configuration objects, paths, scripts, dialog boxes, options, fields, text and list boxes, operational modes, all buttons (including radio buttons), check boxes, commands, tabs, CTI events, and error messages. • The values of options. • Logical arguments and command syntax. • Code samples. Also used for any text that users must manually enter during a configuration or installation procedure, or on a command line.	Select the `Show variables on screen` check box. In the `Operand` text box, enter your formula. Click `OK` to exit the `Properties` dialog box. T-Server distributes the error messages in `EventError` events. If you select `true` for the `inbound-bsns-calls` option, all established inbound calls on a local agent are considered business calls. Enter `exit` on the command line.
Square brackets ([ ])	A particular parameter or value that is optional within a logical argument, a command, or some programming syntax. That is, the presence of the parameter or value is not required to resolve the argument, command, or block of code. The user decides whether to include this optional information.	`smcp_server -host [/flags]`
Angle brackets (< >)	A placeholder for a value that the user must specify. This might be a DN or a port number specific to your enterprise. **Note:** In some cases, angle brackets are required characters in code syntax (for example, in XML schemas). In these cases, italic text is used for placeholder values.	`smcp_server -host <confighost>`

Index

Symbols

A

B

C

D

E

F

G

H

I

L

M

O

P

R

S

T

U

V

W

X