



intelligent Workload Distribution 8.0

Deployment Guide

The information contained herein is proprietary and confidential and cannot be disclosed or duplicated without the prior written consent of Genesys Telecommunications Laboratories, Inc.

Copyright © 2010–2012 Genesys Telecommunications Laboratories, Inc. All rights reserved.

About Genesys

Genesys is the world's leading provider of customer service and contact software - with more than 4,000 customers in 80 countries. Drawing on its more than 20 years of customer service innovation and experience, Genesys is uniquely positioned to help companies bring their people, insights and customer channels together to effectively drive today's customer conversation. Genesys software directs more than 100 million interactions every day, maximizing the value of customer engagement and differentiating the experience by driving personalization and multi-channel customer service - and extending customer service across the enterprise to optimize processes and the performance of customer-facing employees. 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 and the Genesys logo are registered trademarks of Genesys Telecommunications Laboratories, Inc. All other company names and logos may be trademarks or registered trademarks of their respective holders. © 2012 Genesys Telecommunications Laboratories, Inc. All rights reserved.

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 14](#). 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_dep_02-2012_v8.0.003.00



Table of Contents

List of Procedures	9
Preface	11
About intelligent Workload Distribution	11
Support for eServices and Third Party Media Servers	12
Support for Workforce Management.....	13
Intended Audience.....	13
Making Comments on This Document	13
Contacting Genesys Technical Support.....	14
Document History	14
New in Document Version 8.0.003.00	14
New in Document Version 8.0.002.00	15
Chapter 1 intelligent Workload Distribution Overview	17
intelligent Workload Distribution Features	17
Capturing Tasks from Multiple Sources	17
Prioritization of Tasks	19
Reprioritization of Tasks.....	19
Task Distribution	20
Reporting	21
Centralized Logging	21
Technical Licensing.....	21
intelligent Workload Distribution Architecture	22
Task Life Cycle.....	24
New in This Release.....	24
iWD Business Process (IWDBP)	25
Chapter 2 Deployment Overview	27
Installation Overview	27
Application Installation	28
Database Preparation	28

	System Configuration Overview	29
	Tenants	29
	Modules and Components	30
	Solutions	30
	Runtime Nodes	30
	Services	31
	Business Logic Configuration Overview	31
	Departments and Processes.....	31
	Rules.....	32
	Task Classification	33
	Task Prioritization.....	34
	Business Calendars	35
Chapter 3	Installation.....	37
	Task Summary: Installing iWD 8.0	37
	Preparing for Installation.....	39
	Configuration of Application Servers	39
	Installation Prerequisites	40
	CD Structure	41
	Interaction Server Installation	42
	iWD Runtime Node Installation.....	46
	iWD Manager Installation	49
	iWD Configuration Database Preparation.....	50
	iWD Manager Installation.....	51
	Enabling ADDP Connections Between iWD and the Genesys Suite	56
	Enabling/Disabling ADDP for iWD Manager Application	57
	Enabling/Disabling ADDP for iWD Runtime Node	57
	Enabling/Disabling ADDP for BCMS	58
	Installing iWD Rules, iWD Capture, and iWD Data Mart	58
	Post-Installation Steps When Using WebSphere Application Server..	63
	Importing iWD Configuration XMLs	64
	Creating the Tenant and Solution	65
	iWD Setup Utility.....	67
	Installation of iWD Stat Extensions	77
	Stat Server Configuration Options	78
	Link with Genesys Configuration Server	79
	Logging.....	82
	Interaction Server Databases	85
	Installing iWD on UNIX-Based Operating Systems	87
	Sample Application.....	94

Chapter 4

Configuration	97
iWD Manager Overview.....	97
General Conditions for Configuring an iWD Manager User	97
iWD Manager Login	101
iWD Manager User Interface	103
Header Bar	104
Navigation Panel.....	104
Details View	107
Unsaved Changes	109
Context History	109
User Profile	110
General Configuration	111
Profile.....	112
Import and Export	114
Security Policies	116
Managed Tenant Details	118
Lookup Tables.....	120
History.....	123
Modules and Components.....	125
Modules	126
Rule Templates	127
Metrics Templates	135
Solutions and Services	136
Solution Details	137
Runtime Nodes	138
Services	139
Service Details	140
Deployment.....	143
Change History	144
iWD Services	146
Core iWD Services	146
Capture Points	147
Reporting	148
Workforce Management.....	150
Recommended Order of Configuration for iWD Services	150
Service Log Levels	152
Logging Service	152
Configuration Server Connector Service	153
Interaction Server Connector Service	154
Scripting Service	156
Web Service Capture Point.....	157
XML File Capture Point.....	159
Database Capture Point.....	161

	Rules Service.....	167
	Business Context Management Service	168
	Database Service.....	169
	Statistics Adapter	170
	Data Mart ETL Services.....	173
	Configuration of iWD Current-Day Statistics in CCPulse+	178
	Workforce Management Connector Service	180
	Departments and Processes	182
	Department Details	183
	Process Details	185
	Business Calendars	186
	Rules.....	191
	Metrics	203
	Deployment.....	205
	Solution User Access.....	205
Chapter 5	iWD Manager and Task Management	207
	Task Management	207
	Task Monitoring.....	207
	Task Operations	213
	Filters	215
Chapter 6	High Availability.....	219
	Redundancy	219
	High Availability for iWD Components	220
	Redundancy Support Matrix for iWD Services	220
	Configuring Multiple Business Context Management Services	221
	Primary/Backup Configuration	221
	Load Balancing Configuration.....	222
	Deploying Services on Multiple Runtime Nodes.....	222
	Interaction Server Redundancy	223
Appendix A	iWD Messages	225
	Messages	225
	Data Types.....	226
	Task Action	226
	Task Notification.....	227
	CreateTask.....	228
	TaskCreated.....	230
	Error	230
	GetTaskInfo.....	231
	TaskInfo	231

	UpdateTask	236
	TaskUpdated	237
	TaskDistributedQueue	238
	TaskAssigned	239
	CompleteTask	239
	TaskCompleted	239
	HoldTask	240
	TaskHeld	240
	TaskErrorHeld	241
	ResumeTask	241
	TaskResumed	241
	RestartTask	242
	TaskRestarted	242
	CancelTask	243
	TaskCanceled	243
	TaskRejected	244
	Ping.....	244
	Pong	244
	Task Attributes and Interaction Properties.....	245
	Interaction Custom Properties	248
Appendix B	iWD Business Process (IWDBP)	251
	Configuration of List Objects	252
	iWD Business Process	253
	Classification Strategy	256
	Prioritization Strategy	259
	Distribution Strategy	263
	Archive Strategy	268
	Modifying the iWD Business Process	272
Appendix C	Adapting the iWD Business Process for Standard Genesys Channels	275
	Adapting IWDBP to Serve E-mail Interactions	275
	Adding Required Properties to Interactions	275
	Examples	281
	Example 1	283
	Example 2—Departments and Processes	291
	Example 3—How to Add Another Solution	300
Appendix D	iWD Rules and Existing Business Processes	307
	Using iWD Rules in Existing Business Processes.....	307
	Example.....	309

Supplements	Related Documentation Resources	315
	Document Conventions	318
Index	321



List of Procedures

Installing Interaction Server and its databases.	43
Installing the iWD Runtime Node	46
Associating a Person's account with the iWD Runtime Node application	49
Creating the iWD Configuration database	50
Installing iWD Manager	51
Enabling/disabling ADDP for the iWD Manager application.	57
Enabling/disabling ADDP for the iWD Runtime node.	57
Enabling/Disabling ADDP for Business Context Management Services	58
Installing iWD Rules	59
Installing iWD Capture	60
Installing iWD Data Mart.	61
Creating the iWD Data Mart Database.	62
Building WAR archives for iWD Manager and iWD Runtime Node	63
Import default configuration XMLs into iWD Manager	64
Creating an iWD Tenant and iWD Solution in iWD Manager	65
Installing the iWD Setup Utility	68
Using the iWD Setup Utility	69
Installing the iWD Stat Extensions	78
Creating services in iWD Manager.	80
Configuring logging for iWD Manager and iWD Runtime Node	82
Updating the Interaction Server databases	85
Configuring Distribution Points as Lookup Tables	121
Adding a Rule Template to your Solution.	128
Configuring CCPulse+ for iWD.	178
Creating a CCPulse+ template.	179
Configuring iWD to work with WFM	180
Configuring a business calendar	191
Specifying a linear rule	193

Defining a decision table	194
Deploying Services on multiple Runtime Nodes.	223
Configuring a custom interaction property	249



Preface

Welcome to the *intelligent Workload Distribution 8.0 Deployment Guide*. This document describes how to install and configure intelligent Workload Distribution (iWD).

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:

- [About intelligent Workload Distribution, page 11](#)
- [Intended Audience, page 13](#)
- [Making Comments on This Document, page 13](#)
- [Contacting Genesys Technical Support, page 14](#)
- [Document Change History, page 14](#)

For information about related resources and about the conventions that are used in this document, see the supplementary material starting on [page 315](#).

About intelligent Workload Distribution

intelligent Workload Distribution (iWD) is an application that works with the Genesys Customer Interaction Management (CIM) Platform to distribute tasks to the resources that are 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 intraday and historical status of the tasks and the task handling.

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

iWD provides a user interface that is designed specifically for business users—giving them access to not only 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 that has a specific set of attributes.

Support for eServices and Third Party Media Servers

The iWD 8.0 components work together with Interaction Server to make up the iWD Solution. Interaction Server is an integral component for iWD, whereas formerly it was solely a component of the Genesys eServices solution (formerly called Multimedia). iWD 8.0 uses the Interaction Server database to store task information, whereas previous releases of iWD used a separate iWD runtime database.

Note: iWD, Interaction Server, E-mail Server (for outbound notifications and acknowledgements), and Knowledge Management together make up the iWD *solution*. The iWD *application* refers to the software components that are packaged on the iWD CDs, such as iWD Runtime and iWD Manager. Throughout this document, the iWD solution will be referenced. Remember that this solution shares some common components with the Genesys eServices solution, such as Interaction Server and, optionally, Genesys Knowledge Management and Genesys E-mail.

iWD 8.0 can be used with Genesys eServices solutions (for example, Genesys E-mail, Genesys Chat, and Genesys SMS) as well as with integrations to third party media servers that were built by using the Open Media API. When used together, these combined solutions allow an enterprise to apply business rules to any interaction that is managed through the Genesys Interaction Server, such as e-mail, chat, and SMS interactions. Moreover, these interactions can be managed through iWD Manager’s Global Task List—allowing a business analyst to view the status of these interactions, hold/resume the interactions, and modify various attributes of the interactions. See also “iWD Business Process (IWDBP)” on [page 25](#), and Appendix B, “iWD Business Process (IWDBP),” on [page 251](#).

Support for Workforce Management

Beginning in iWD 8.0, the Standard Rules Template now enables you to specify the WFM Activity or Multi-Site Activity to assign to a task as part of a business rule. The Assign WFM Activity action has been added to the template.

For example, the following conditions and actions could be configured for a Rule:

```
if the Product is Widget, and the Customer Segment is Gold, then Set
Priority to 200 and Set WFM Activity to 'Gold Product Support'
```

This sets a piece of attached data for the interaction that has the name of the WFM Activity or Multi-Site Activity. Then, in the routing strategy, you can use that piece of attached data for segmentation to peg the interaction to a specific Interaction Queue or Virtual Queue (the object types that WFM Data Aggregator is capable of monitoring). Refer to “Workforce Management Connector Service” on [page 180](#) and the Workforce Management documentation for more information.

Intended Audience

This document is primarily intended for IT staff who are responsible for the iWD installation and configuration and business analysts who are responsible for department and process configuration. It has been written with the assumption that you have a basic understanding of:

- The workflow concepts as implemented in the various enterprise source systems (for example, business process management (BPM) systems, host systems, CRM systems, and so on) from which iWD will capture tasks.
- 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 (international)	support@genesyslab.com.au
Malaysia	1-800-814-472 (toll-free) +61-7-3368-6868 (international)	support@genesyslab.com.au
India	000-800-100-7136 (toll-free) +61-7-3368-6868 (international)	support@genesyslab.com.au
Japan	+81-3-6361-8950	support@genesyslab.co.jp
Before contacting technical support, refer to the <i>Genesys Technical Support Guide</i> for complete contact information and procedures.		

Document History

Document Change History

This section will list topics that are new or that have changed significantly since the first release of this document.

New in Document Version 8.0.003.00

This document has been updated to support the latest release of iWD. The following topics have been added or significantly changed since document version 8.0.002.00:

- New information about enabling ADDP connections has been added (see “Enabling ADDP Connections Between iWD and the Genesys Suite” on [page 56](#)).
- New information about general conditions for creating an iWD Manager user have been added (see “General Conditions for Configuring an iWD Manager User” on [page 97](#)).
- New information about configuring application servers has been added. See “Configuration of Application Servers” on [page 39](#).

- Figure 3, “Task Life Cycle,” on [page 24](#) has been updated.
- Compatibility information for Genesys Universal Routing Server (URS) has been added.
- Various minor textual corrections have been made throughout the document.
- A note was added to the iWD Data Mart installation procedure indicating that each iWD tenant solution requires its own Data Mart (see [page 61](#)).
- A new bullet was added to the Next Steps of the iWD Data Mart installation procedure that is of relevance for users of the Interactive Insights for iWD product (see [page 61](#)).
- A warning was added to the description of the Rule Language Expression property of conditions and actions (see [page 130](#)).
- The URL for the iWD website has been updated throughout the document. The URL is <https://sites.google.com/a/iwdlab.com/iwd8/>.

New in Document Version 8.0.002.00

This document has been updated to support the latest release of iWD. The following topics have been added or significantly changed since document version 8.0.001.00:

- The procedures for installing iWD Rules, iWD Capture, and iWD Data Mart have been updated. You no longer need to run the installation package twice if iWD Manager and iWD Runtime Node are on the same host.
- The tables that contain Service properties have been updated. New and previously undocumented properties have been added for most iWD Services. See “iWD Services” on [page 146](#).
- A new section has been added to [Chapter 6](#). See “Deploying Services on Multiple Runtime Nodes” on [page 222](#).
- Appendix B, “iWD Business Process (IWDBP),” on [page 251](#), has been updated to reflect the updates that were made to the iWD business process (IWDBP).



Chapter

1

intelligent Workload Distribution Overview

This chapter provides an overview of intelligent Workload Distribution (iWD). It contains the following sections:

- [intelligent Workload Distribution Features, page 17](#)
- [intelligent Workload Distribution Architecture, page 22](#)
- [New in This Release, page 24](#)
- [iWD Business Process \(IWDBP\), page 25](#)

intelligent Workload Distribution Features

intelligent Workload Distribution (iWD) creates an enterprise-wide task list that is centrally managed and prioritized. It allows work to be presented to the right resource, at the right time, and at the right location. It gets non-real-time work (tasks) from multiple source systems, uses business rules to prioritize or reprioritize the tasks, and then distributes the tasks to the most suitable resource.

Capturing Tasks from Multiple Sources

A key function of the iWD solution is the ability to capture (or get) work from the multitude of work sources in the enterprise, such as documentation management systems, CRM systems, workflow systems, claims administration systems, legacy host systems, and so on. iWD integrates with these source systems through **Capture Adapters**. The out-of-the-box Capture Adapters are:

- **Web Service** for use with source systems that have service-oriented interfaces.
- **XML**, for integrating with source systems that include the ability to generate XML files.

- **Database**, for systems that do not offer service-oriented interfaces or provide XML output capabilities.

Note: Beginning in version 8.0.2, eServices offers integrated **JMS** Capture Point functionality. Refer to the eServices documentation for more information. Version 8.0.210 of eServices (specifically, the Interaction Server component) also includes integrated XML File Capture Point functionality. Version 8.1.0 of eServices (the Interaction Server component) includes an integrated Database Capture Point.

Capture Points are enabled by a Capture Adapter. Each Capture Point is a specific instantiation of the Adapter for capturing a specific sort of work, that is often associated with a specific business process, such as an order, refund, or return. The Adapter is technology-specific, while the Capture Point can be specific to a source system, a category of work that is derived from a particular source system, or even a specific business process.

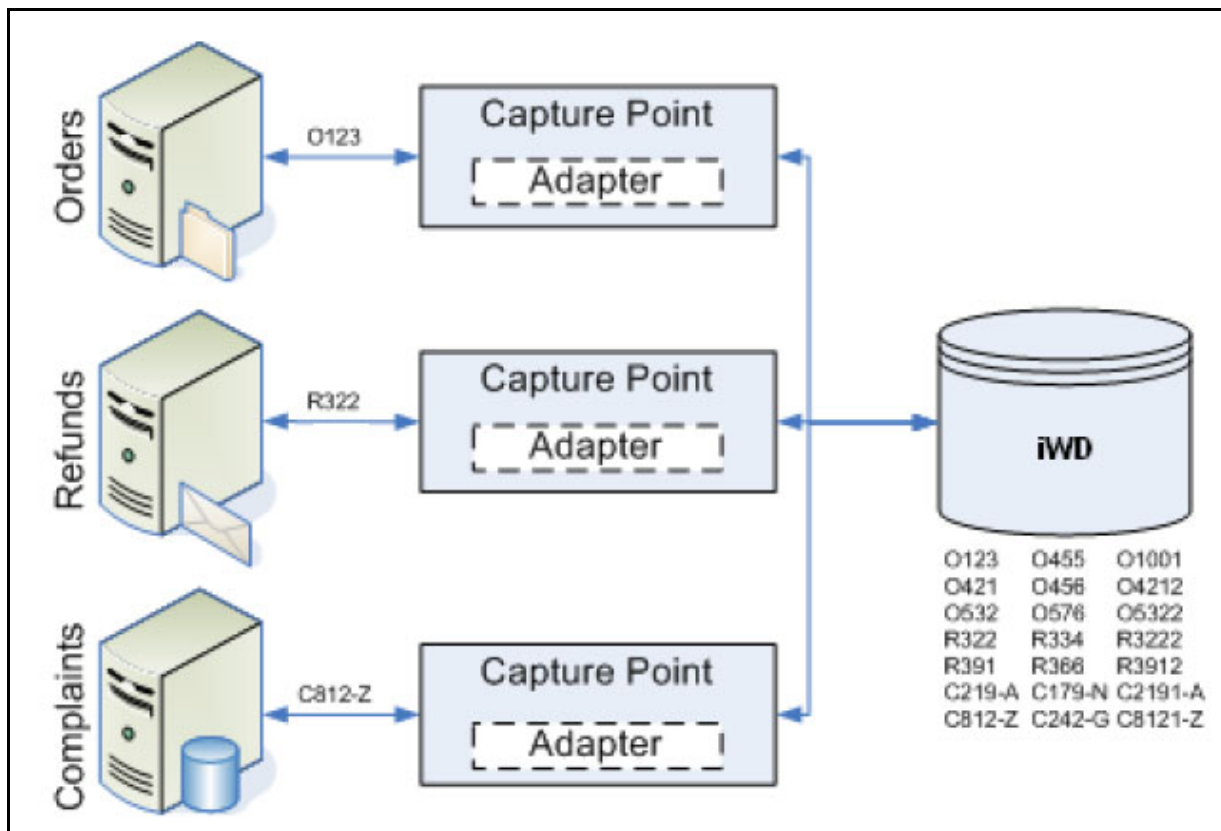


Figure 1: Capture Point Process Diagram

To establish a connection with the correct source system, each Capture Point requires the configuration of specific properties such as file directories for

Capture Points that use the XML Capture Adapter or SQL queries for Capture Points that use the Database Capture Adapter.

Note: In some configurations, the Capture Point is not necessarily specific to a business process. For example, one Capture Point can support capturing orders, billing, and complaints.

Prioritization of Tasks

Prioritization is the process by which iWD arranges the Global Task List in order of priority or importance, based on business rules that are configured at the Global level, or for the Capture Point, Department, and Process. The fulfillment of one task over another might provide a benefit to the business, such as increased revenue, decreased costs, improved customer satisfaction, or avoidance of a penalty or fine.

Business rules within iWD are based on Business Rule Templates that are provided out-of-the-box with the iWD installation, or created by IT personnel. Business rule templates are the foundation upon which users create or modify existing business rules that govern iWD. Business users can be empowered to create or modify rules without involving IT personnel.

Business rules can also be used to facilitate a more informed prioritization decision by leveraging existing enterprise web services to collect additional task-related information. For example, you can configure a rule that, upon the capture of any order, retrieves the customer's current credit score. The score (such as poor) is then attached to the task and can be used downstream by another business rule for setting priority, or it can be used later within a Genesys routing strategy for determining to whom the task should be routed (for example, to a credit-granting department prior to sales).

Reprioritization of Tasks

At any time, the information that is related to a task can change and affect the task's priority. A simple example of where reprioritization can affect the initial priority that is set is the time that remains before the due date of a task. For example, assume that you have a time-sensitive process that includes tasks that involve dispute resolution. If the disputes are not resolved within a specific number of days (for example, 10 days), the organization might be fined. You can configure a business rule that specifies that if such a task is within two days of its due date, the task should be reprioritized with the highest priority, so that it is immediately assigned to an agent.

iWD can be configured so that each captured task receives a task reprioritization interval, when business rules are applied and new values are set for the task. Some tasks might increase in priority, while others might decrease.

All business rules have a unique identifier and description. When a rule is applied, the rule name and value are captured and stored for future business reporting, including business and internal auditing.

Task Distribution

All iWD tasks are managed through the Genesys Interaction Server and are assigned (routed) to agents or knowledge workers by the Genesys Universal Routing Server (URS). A Distribution Point is an optional attribute of a task that gets associated to a task as part of a business rule. It can then be used to help refine how the task is managed by Interaction Server and URS. For example, logic can be written into a routing strategy that uses the Distribution Point attribute of a task to make decisions about the types of resources or locations to which the task should be routed.

The Distribution Point is also propagated to the iWD Data Mart as a dimension that can be used for creating reports.

One example of how Distribution Points can be used is to set up a Distribution Point to represent a contact center and a separate Distribution Point to represent an outsourcer who will handle the tasks that are related to several iWD business processes. The appropriate Distribution Point will get associated to a task as part of the logic of a business rule—thus, attaching the Distribution Point name as an attribute of the task. Within the routing strategy that handles the assignment of the tasks to the knowledge workers, this attribute can be read, assigned to a local strategy variable, and then used to route to the appropriate Agent Group at the right location. Reports can be generated that segment the task handling by the different resource types who handled the tasks (internal contact center versus outsourcer, in this example).

Note: The role of a Distribution Point has changed significantly between iWD 7.6.1 and iWD 8.0. Users of iWD 7.6.1 who are migrating to iWD 8.0 should know that Distribution Points are no longer separate iWD services that define specific Interaction Server queues and thresholds to which iWD tasks will be distributed. Instead, in iWD 8.0, all tasks are immediately distributed to Interaction Server upon task capture, and the Distribution Point is used mainly as a reporting dimension, as well as a way to help segment tasks to guide the assignment that is defined in the routing strategy logic.

Although iWD performs prioritization and reprioritization, it does so only to set values for priority routing within the Genesys CIM Platform. URS can leverage the iWD-calculated priority and business values in its routing strategies, or it might calculate its own; in either case, URS ensures that the most critical tasks are presented to agents first. URS continues to reevaluate priority for tasks that it has received against real-time voice and other non-voice interactions—ensuring that the most important is presented next.

Reporting

iWD monitors the entire lifecycle of tasks, from the moment that they are captured until they are archived and stopped (removed from the system). iWD provides business-activity monitoring of a number of events, including the following:

- **New**—The point at which the task was captured by the Capture Point
- **Process/Reprioritization Rules Applied**—All process rules that are applied to the task, as well as the values that are calculated and assigned to the task
- **Queued**—The point at which the task has been classified and prioritized and is awaiting potential reprioritization or assignment to a resource
- **Assigned**—When the task was assigned to an agent for processing
- **Transfer/Transfer to Queue**—Whether the agent transferred the task to another agent or back to the queue
- **Held**—Whether the task was held (manual hold)

Each record is timestamped and stored in the iWD Data Mart, where the data that is collected can be leveraged in third-party reporting applications.

Note: In addition to the iWD Data Mart, iWD 8.0 interoperates with Interaction Concentrator and Genesys Info Mart for historical reporting. Certain fact tables (for example, `MMEDIA_IXN_FACT_EXT` and `MMEDIA_SEG_FACT_EXT`) store media-specific facts about open media as well as multimedia interactions and multimedia interaction segments. iWD tasks all flow through Interaction Server queues as Open Media interactions, so that these tables are populated with iWD data. Refer to the Interaction Concentrator and Genesys Info Mart documentation for more information.

Centralized Logging

iWD 8.0 also has a centralized logging feature that supports logging of iWD log messages through Genesys Message Server. This feature provides the additional capabilities of viewing these log messages through a centralized log viewer (such as that included in Genesys Administrator) as well as the ability to generate alarms and SNMP traps through Genesys Solution Control Server. Refer to “Logging” on [page 82](#) for information about configuring centralized logging.

Technical Licensing

To view tasks (interactions) in the iWD Global Task List, iWD Manager connects to Interaction Server on behalf of an agent. The user who is logged into iWD Manager must have a Place ID configured, and the connection to

Interaction Server is made on behalf of this user who has this Place ID. The number of iWD Manager users cannot exceed the number of `ics_multi_media_agent_seat` technical licenses that you have provisioned in your FlexLM license file. Keep in mind that Interaction Server also uses agent seat licenses for agents who will be accepting and processing tasks (interactions) at their agent desktops, so that the total number of `ics_multi_media_agent_seat` technical licenses that you have should account for not only the number of concurrent agents who are processing tasks at their desktops, but also the number of concurrent users who will be accessing the Global Task List.

To process the iWD tasks at the agent desktop, Interaction Server also checks the number of licenses that are provisioned in the FlexLM license file for each media type that is being handled by the agents. These media type technical licenses are not required for any iWD Manager usage.

If you plan to use iWD to process e-mails that are not being “captured” through the Genesys E-mail Server, then it is best to use a custom media type, rather than the media type `email`. This is because handling the media type `email` requires a specific FlexLM license to be checked out (`ics_email_webform_channel`), whereas there is a separate general FlexLM license used for all custom media types (`ics_custom_media_channel`). For this reason, it is recommended to create a custom media type (such as `email1`) for work items of type “email” that are not being captured by the Genesys E-mail Server.

Please refer to the *Genesys Licensing Guide* for more information.

intelligent Workload Distribution Architecture

The core iWD solution building blocks are implemented as Java services, which can run on several different Java application servers. iWD does not rely on any J2EE-specific functions.

Figure 2 illustrates the main intelligent Workload Distribution building blocks (services), the relationships among them, and the external components that are involved.

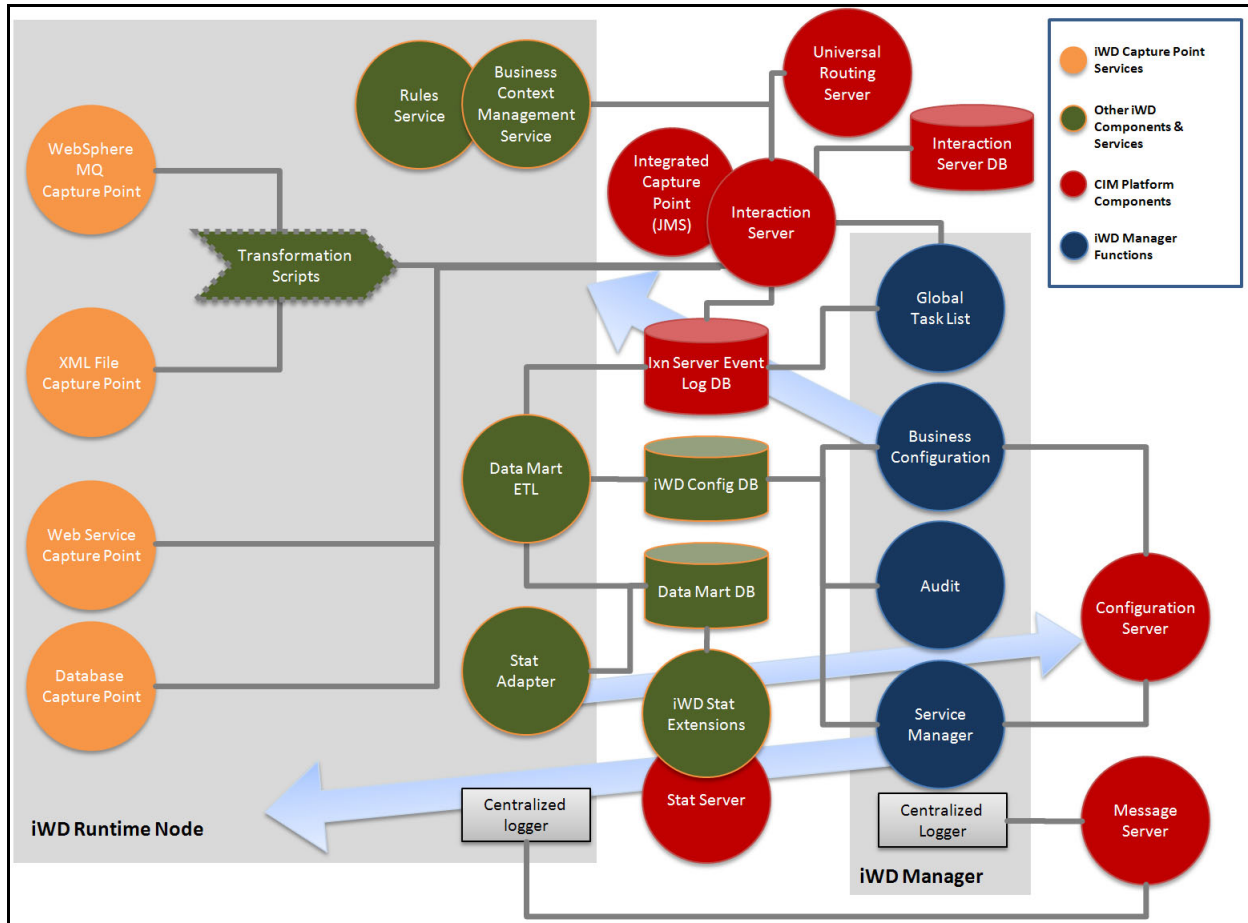


Figure 2: intelligent Workload Distribution High-Level Architecture

Task Life Cycle

Figure 3 illustrates the overall task life cycle in iWD:

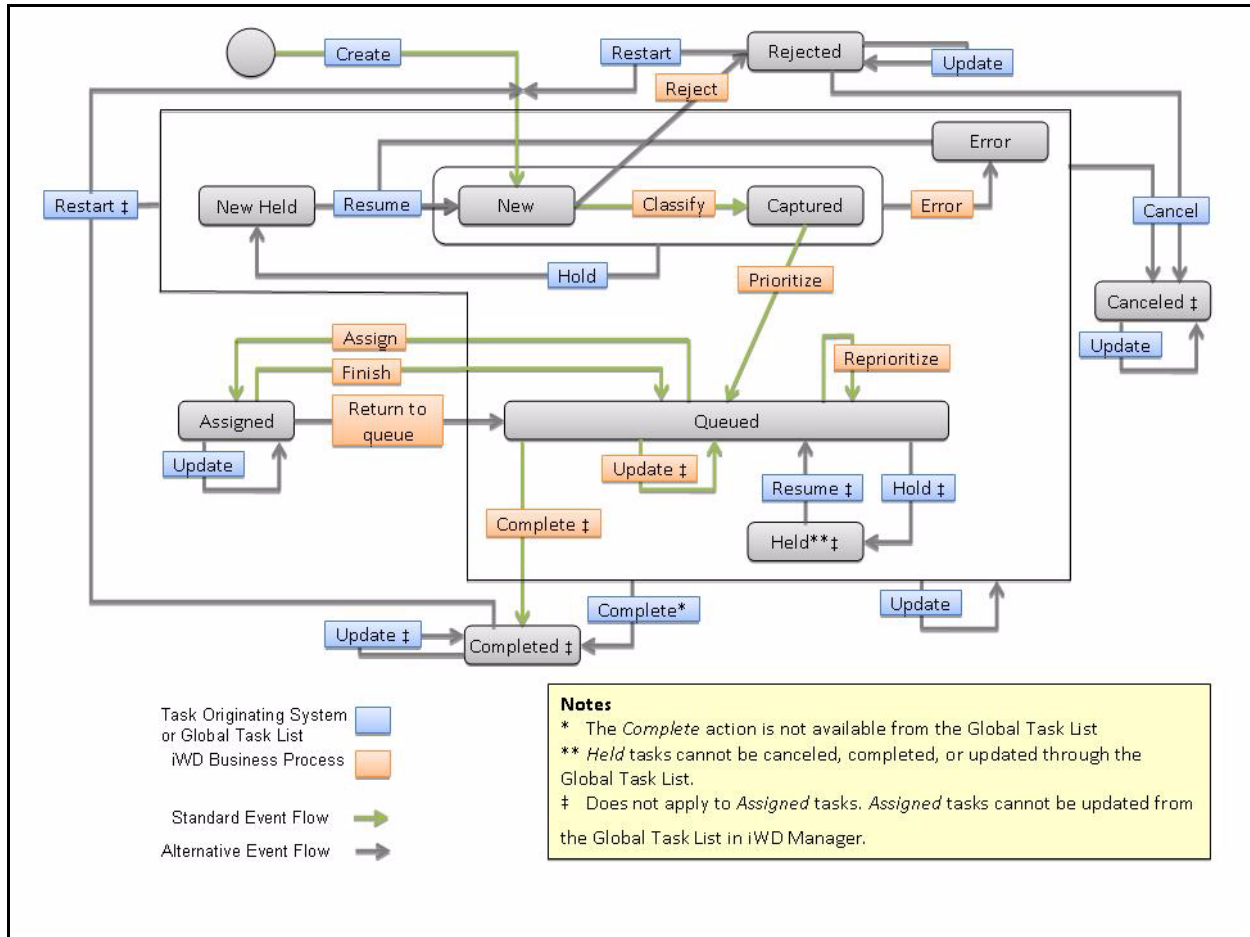


Figure 3: Task Life Cycle

New in This Release

iWD 8.0 includes the following new features:

- iWD now stores all tasks in the Interaction Server database, instead of an independent iWD runtime database.
- iWD no longer relies on T-Server to propagate statistics from iWD to Stat Server.
- iWD now uses Configuration Server authentication.
- The iWD rules service is invoked as an External server from within a routing strategy, which opens up the ability for business processes that are using Genesys eServices media servers such as Genesys E-mail, Genesys Chat, and Genesys SMS to use business rules for classification and prioritization.

- New OS support for server components:
 - AIX 6.1
 - Windows Server 2008
 - Red Hat Enterprise Linux 5
 - Solaris 10
- New database support
 - Oracle 10g and 11g RAC
 - Microsoft SQL Server 2005 and 2008
- New web-browser support:
 - Microsoft Internet Explorer 8
- Internationalization
- Localization kit
- End-to-end sample application

iWD Business Process (IWDBP)

iWD 8.0 is packaged with an out-of-the-box Business Process. The Business Process is installed with the iWD Setup Utility and is made up of a set of Interaction Queues that map to the iWD state model (NEW, ERRORHELD, CAPTURED, COMPLETED, CANCELED, REJECTED, and QUEUED). From this Business Process, interactions are submitted to routing strategies. From within a routing strategy, the External Service Protocol (ESP) block is used to call the iWD Rules Service. This approach is used to apply classification and prioritization rules to the interaction. When a user goes to the Global Task List view in iWD Manager, to monitor the interactions that are in various states, this component will communicate with Interaction Server in order to retrieve the list of interactions and their attributes.

This out-of-the-box Genesys iWD Business Process maps to the iWD state model, allowing you to use iWD-based reporting for other interaction types (for example, you might want to track Genesys e-mails along with other task types, under the same Department or Process).

This Genesys iWD Business Process is completely optional for iWD customers who are using Genesys E-mail, Genesys Chat, Genesys SMS, or even third-party e-mail, SMS, or chat. For Genesys eServices customers, the Genesys iWD Business Process can be left unchanged if you want to use business rules only. In this scenario, what would change would be the routing strategies. The strategies would use the ESP block to invoke the iWD business rules service.

This means that existing Genesys E-mail, Chat or SMS/MMS customers can use the business rules within iWD without having to change their Genesys Business Processes; or, to access some additional functionality, changes can be made to the Business Processes.

For a detailed description of the iWD Business Process, including its strategies, refer to Appendix B on [page 251](#).



Chapter

2

Deployment Overview

This chapter provides an overview of iWD deployment.

This chapter contains the following sections:

- [Installation Overview, page 27](#)
- [System Configuration Overview, page 29](#)
- [Business Logic Configuration Overview, page 31](#)

Installation Overview

Installation is the initial iWD deployment phase that results in a fully functional iWD Manager application and prepared iWD runtime nodes. iWD Manager will be used for the rest of the deployment configuration, while runtime nodes are containers in which the iWD runtime services will run.

The installation phase requires knowledge of the overall system infrastructure in which iWD is being deployed, such as Java application server and database configuration. This phase should typically be implemented by IT personnel.

iWD installation consists of two steps:

- “Application Installation” on [page 28](#).
- “Database Preparation” on [page 28](#).

Note: For detailed installation information, see Chapter 3, “Installation,” on [page 37](#).

Compatibility with URS

iWD 8.0 is *not* compatible with URS 7.5.

Application Installation

After the iWD operational databases have been prepared, the iWD Manager application and iWD runtime nodes can be installed. Both of these components run on a Java application server.

Note: Refer to the *Genesys Supported Operating Environment Reference Manual* for a list of the Java application servers that are supported in iWD.

A basic iWD deployment utilizes a single instance of iWD Manager, as well as a single instance of an iWD runtime node. In more complex scenarios, such as multi-tenant, high-volume or high-availability deployments, multiple iWD runtime node instances can be installed. Such deployments provide more controlled resource partitioning and allow load distribution across multiple physical servers.

Database Preparation

The first step of iWD installation is the database preparation. A basic iWD deployment utilizes two operational databases:

- iWD Configuration database—Stores iWD system and business logic configuration such as services, processes, and rules (see Chapter 4, “Configuration,” on [page 97](#) for descriptions of these objects).
- Interaction Server database—iWD tasks are stored in the Interaction Server database. Therefore, make sure that you have installed and configured the Interaction Server and its associated database, as described in the *eServices (Multimedia) 8.0 Deployment Guide*.

Note: For a list of the database engines that are supported by the iWD application and Interaction Server, refer to the *Genesys Supported Operating Environment Reference Manual*. Please note that the iWD might not support all of the databases that are supported by Interaction Server.

Preparation of the iWD Configuration database requires that actions be performed manually via the database’s administrative interface:

1. Creation of a database
2. Creation of a database user account or accounts

The rest of the database setup, such as creation of tables and indexes, is performed automatically by iWD. Alternatively, if your database administrator prefers to inspect the SQL statements prior to the database initialization and table creation, iWD includes a command-line utility for this. See “iWD Configuration Database Upgrade Utility (dbup)” on [page 55](#).

System Configuration Overview

iWD system configuration results in a fully set-up iWD system infrastructure that consists of tenants, solutions, and services that enable iWD functionality.

The iWD system configuration phase requires knowledge of the overall system infrastructure in which iWD is being deployed, such as Java application server and database configuration.

The following topics provide an overview of different system configuration aspects:

- “Tenants” on [page 29](#)
- “Solutions” on [page 30](#)
- “Services” on [page 31](#)

For a detailed system configuration reference, see Chapter 4, “Configuration,” on [page 97](#)—specifically, the following sections: “General Configuration” on [page 111](#), “Modules and Components” on [page 125](#), and “Solutions and Services” on [page 136](#).

Tenants

iWD configuration supports multi-tenancy. iWD automatically creates a root tenant that is named System by default. Although it is possible to configure iWD solutions, services, and business logic directly at the System tenant level, it is recommended that you create a subtenant for that configuration. In iWD terminology, such a subtenant is called a managed tenant. This managed tenant is associated directly with a tenant defined in Genesys Configuration Server. In a single-tenant environment the configured managed tenant must map to the Resources tenant in Genesys Configuration Server.

A user who is configuring the system in iWD Manager can have access to one or more tenants. Access is defined by the security policy that is configured per tenant. The policy allows definition of an arbitrary numbers of user roles, where each role is mapped to a single Genesys Configuration Server Access Group and has a number of associated iWD permissions. The actual permissions that a user has are then determined based on the Configuration Server Access Groups to which the user belongs.

The modules that are accessible to the tenant are managed by the parent. A module by itself represents specific iWD functionality, such as “XML Capture” or “iWD Rules.” The specific functionality is implemented as a collection of components that can be services or business logic templates (such as rules templates).

Note: The tenancy model that is supported in the iWD application currently has a one-to-one relationship with the tenancy model in Genesys Configuration Server.

Multi-tenant Configurations If you are using a Multi-Tenant Configuration Server, please be aware that the iWD tenant can only access resources (such as Skills and Agent Groups) that are specifically configured under the corresponding Genesys Configuration Server Tenant in Configuration Manager or Genesys Administrator. Therefore, any Skills, Agent Groups, or other resources that are configured at the Environment Level will not be accessible in rules at a child-tenant level in iWD. To make these resources accessible, they must be configured as resources under the child tenant in Configuration Server.

Modules and Components

Modules and components define iWD functionality:

- A component is an atomic object that provides a specific iWD function. The following component types are used in iWD:
 - Service template: This component represents an iWD service that implements specific functionality. Service templates are preconfigured and thus cannot be changed in iWD Manager. Service instances that are based on service templates, however, are configurable.
 - Rule template
- A module is a group of components.

Solutions

Solutions are used for partitioning logical and physical resources for purposes of user access control and load partitioning (performance). Normally there will be one Solution per Tenant.

A solution in iWD represents a runtime environment, which is composed of the following:

- Runtime nodes (see [page 30](#))—iWD runtime application instances that are within the Java application server in which services are being run
- Services (see [page 31](#))—Services that enable iWD functionality, such as capture, rules, and logging
- Business logic configuration (see [page 31](#))—iWD departments, processes, rules, and more.

Multiple solution instances can be configured per tenant, if necessary (for example, “Production” and “Test”).

Runtime nodes require simple preparation during installation, as described in “Installation Overview” on [page 27](#).

Runtime Nodes

Runtime nodes are containers for iWD services that enable the necessary service management infrastructure. Physically, runtime nodes are instances of the iWD runtime application and are run within a Java application server.

Services that run within a runtime node are configured in (and managed through) the iWD Manager application.

Services

iWD Services implement actual iWD functionality, such as capturing tasks or loading data into the Data Mart. Refer to “iWD Services” on [page 146](#) for more information about iWD Services, including the recommended order of configuration.

Business Logic Configuration Overview

The iWD business logic configuration phase is where iWD business context is introduced. This includes definition of departments, processes, business rules, and business calendars. After this phase, iWD is fully functional and can start processing tasks.

The iWD system configuration phase requires knowledge of business context for tasks that will be handled by iWD. This includes business processes, service-level agreements (SLAs), and other factors that influence task-handling logic.

The following topics provide an overview of different business configuration aspects:

- “Departments and Processes” on [page 31](#)
- “Rules” on [page 32](#)
- “Task Classification” on [page 33](#)
- “Task Prioritization” on [page 34](#)
- “Business Calendars” on [page 35](#)

For a detailed business configuration reference, see the section “Departments and Processes” on [page 182](#) of [Chapter 4](#).

Departments and Processes

Departments represent enterprise departments for which iWD will perform task prioritization and routing. Processes represent the business processes that are within those enterprise departments. In iWD, processes are always grouped within (associated with) a department. Departments and Processes allow for the definition of task-handling business rules that are specific to a department or process context. For more information about rules, see both “Task Classification” on [page 33](#) and “Task Prioritization” on [page 34](#).

Each department and process also allows for the definition of an arbitrary number of custom attributes that provide additional enterprise-specific context for reporting purposes. [Figure 4](#) shows an example of a process.

ACME > Departments > Sales Department > Complaint

General Rules

ID ACME_SD_Cmp **Process Name** Complaint **Start Date** 01/01/2007 **End Date** 31/12/2010

Description

Process Complain for Sales Department/ Customer Service Department.
KPI's defined below - need to checked at the end of each month.

Contact Name Frank Miller **Contact Email** +1 415 1234567890 **Contact Phone** miller@acme.com

Custom Process Attributes

Name	Type	Value	Description	
Costs per task	Currency	4.50	Cost per complain (Internal Full Cost)	+ -
KPI: SLA	Number	1440	Contractual agrred Service Level (First Handling)	+ -
KPI: SLA total	Number	10080	Contractual agreed Service Level (Start - End)	+ -
KPI: avg Process time	Number	15	Contractual agreed Process Handling time (Agent working or	+ -

Save Save & Close Cancel Delete View tasks

Figure 4: Example Process

Rules

As part of configuring departments and processes in iWD, you will configure and associate rules. These rules will define the task-handling business logic that is applicable to the departments and/or processes. Generally, a rule is represented by one or more conditions and one or more actions. If all of the conditions are true, all of the actions will be executed. If any condition is not true, none of the actions will be executed.

Rules are expressed in an easy-to-understand human language, such as, “If task is due in 10 or more minutes, increase priority by 10.” The implementation details are hidden in rule templates, and users who configure business logic deal only with high-level logical expressions.

Rules can be defined in one of two ways: as a linear rule or as a decision table. Linear rules are intended for complex rules that have many conditions. Each

condition or action is represented by a single line in the rule. [Figure 5](#) shows an example of a linear rule.

Expression	Parameters
Due Time is in	.0 to .2 .minutes.
Increase Priority	.10.

Figure 5: Linear Rule Example

Decision tables represent a more compact form of rule representation; however, they might not be as well suited to complex rules. In a decision table, multiple rules are grouped together, so that each condition or action is represented by a column in a table, and each row represents a rule. The number and type of conditions and actions (columns) is constant across all of the rules in the list. [Figure 6](#) shows an example of a decision table.

ID	Name	Due Time is in	Increase Priority	Reprioritize after
DT_1_1	Due in 0 to 2 minutes	.0 to .2 .minutes.	.90.	.1 working .minutes.
DT_1_2	Due in 2 to 4 minutes	.2 to .4 .minutes.	.50.	.1 working .minutes.
DT_1_3	Due in 4 to 999 minutes	.4 to .999 .minutes.	.10.	.3 working .minutes.

Figure 6: Decision Table Example

Task Classification

The primary purpose of task classification is to associate a task with a configured process. Additionally, classification can assign different task attributes, such as business value and due time.

Task-classification logic is expressed via business rules that can be defined for four different contexts:

- Global Rules
- Capture Point
- Department
- Process

If rules are defined for more than one context, they are applied in sequence, as previously listed. After a process has been selected for a task, additional rules are applied that have been defined for the selected process and its associated department. [Figure 7](#) shows an example of Task Classification.

ID	Name	Phase	Start Date
LR_7	LR2	classification	Jul 8, 2010
DT_1	Classify	classification	26/08/2010

ID	Name	Process is	Set business value
DT_1_1	Complaint	Complaint	100
DT_1_2	Address Change	Address Change	10

Figure 7: Task Classification

For more information about task classification, refer to Appendix B on [page 251](#), specifically “Classification Strategy” on [page 256](#).

Task Prioritization

The primary purpose of task prioritization is dynamic priority calculation, where dynamic means that the task priority can be recalculated multiple times during the task’s life cycle. As with task classification, prioritization logic is expressed via rules.

Prioritization rules are initially applied immediately after department-/process-level classification rules and then reapplied after a specified reprioritization period. The reprioritization period is expressed in the same way as any other rule action, as illustrated in [Figure 8](#).

ID	Name	Phase	Start Date	End Date
DT_1	Reprioritize	prioritization	26/08/2010	

ID	Name	Due Time is in	Increase Priority	Reprioritize after
DT_1_1	Due in 0 to 2 minutes	0 to 2 minutes	90	1 working minutes
DT_1_2	Due in 2 to 4 minutes	2 to 4 minutes	50	1 working minutes
DT_1_3	Due in 4 to 999 minutes	4 to 999 minutes	10	3 working minutes

Figure 8: Task Prioritization

If a reprioritization period is not set for a task during the prioritization phase in business rules, the `IWD_reprioritizeDateTime` attribute is set to Dec 31, 2030. Therefore, for all intents and purposes, the task will not undergo further reprioritization unless it is restarted.

The Standard Rules Template includes two rule conditions that should be used in prioritization rules to ensure that the reprioritization interval is set correctly, while avoiding any unnecessary immediate reprioritization of a task (that is, the first time prioritization rules are evaluated).

For example, suppose you have a task that, during the classification phase, gets an initial priority of 100. You wish to increase the priority by 10 every 1 hour, if the task is due in less than 24 hours. You want to do the first check 10 minutes after the task is classified. You would set this up by using two different prioritization rules, configured in the order shown below. [Figure 9](#) shows the first rule, which includes the `Is first prioritization` condition. The second rule, shown in [Figure 10](#), includes the `Is reprioritization` condition.

ID	Name	Due Time is in	Is first prioritization	Reprioritize after
RULE_70	Set initial reprioritization time	0 to 24 hours		10 minutes

Figure 9: Rule 1—Is First Prioritization

ID	Name	Due Time is in	Is reprioritization	Increase Priority	Reprioritize after
RULE_69	Reprioritize tasks due in 0 to 24 hrs	0 to 24 hours		10	1 hours

Figure 10: Rule 2—Is Reprioritization

For more information about task prioritization, refer to Appendix B on [page 251](#), specifically “Prioritization Strategy” on [page 259](#).

Business Calendars

A business calendar is a set of configuration parameters that define working days and hours, as well as holidays that apply to the business. In its simplest form, the business calendar would consist of definitions for both a working week and working hours that apply to all working days. A definition of a working week from Monday to Friday—in which each day starts at 9:00 AM and ends at 5:00 PM—is a classic example of a simple business calendar. If necessary, exceptions to the usual working schedule (public holidays, business-specific holidays, nonstandard working hours, and so on) can be added to the business calendar. Business calendars can be used in iWD rules to perform date and time calculations that take into account the working schedule of the business. Business Calendars have to be assigned by a rule, and this rule

must run before other rules that use the business calendar. [Figure 11](#) shows a sample business calendar.

> Business Calendars

Name
Day Shift

New Business Calendar

Day Shift

ID: BC0 Name: Day Shift

Timezone: America/New_York (GMT-5) (+DST)

Week starts on: Monday Week ends on: Friday Start time: 09:00 End time: 17:00

Business calendar rules

Name	Entry type	Calendar placement	Definition
New Year's Day	Holiday	Fixed	01/01/2010

Save Save & Close Cancel Copy

Figure 11: Business Calendar

3

Installation

This chapter provides information about installing iWD.

This chapter contains the following sections:

- [Task Summary: Installing iWD 8.0, page 37](#)
- [Preparing for Installation, page 39](#)
- [Interaction Server Installation, page 42](#)
- [iWD Runtime Node Installation, page 46](#)
- [iWD Manager Installation, page 49](#)
- [Enabling ADDP Connections Between iWD and the Genesys Suite, page 56](#)
- [Installing iWD Rules, iWD Capture, and iWD Data Mart, page 58](#)
- [Importing iWD Configuration XMLs, page 64](#)
- [Creating the Tenant and Solution, page 65](#)
- [iWD Setup Utility, page 67](#)
- [Installation of iWD Stat Extensions, page 77](#)
- [Link with Genesys Configuration Server, page 79](#)
- [Logging, page 82](#)
- [Interaction Server Databases, page 85](#)
- [Installing iWD on UNIX-Based Operating Systems, page 87](#)
- [Sample Application, page 94](#)

Task Summary: Installing iWD 8.0

The following table outlines the task flow for installing iWD 8.0. The procedures in this table provide instructions about installing iWD components on Windows hosts. For information about installing on Unix-based operating systems, refer to “Installing iWD on UNIX-Based Operating Systems” on [page 87](#).

Task Summary: Installing iWD 8.0

Objective	Related Procedures and Actions
1. Prepare for installation and review prerequisites.	<ul style="list-style-type: none"> Ensure that your environment meets the prerequisites that are outlined in “Installation Prerequisites ” on page 40. Ensure that the required CDs are available. See “CD Structure” on page 41.
2. Install Interaction Server	iWD 8.0 requires Interaction Server 8.0.1 or higher. If your environment does not have an instance of Interaction Server installed already (such as in an eServices solution), you must install one. See Procedure: Installing Interaction Server and its databases , on page 43 .
3. Install iWD Runtime Node and associate a Person account to the iWD Runtime Node application.	<ul style="list-style-type: none"> Procedure: Installing the iWD Runtime Node, on page 46. Procedure: Associating a Person’s account with the iWD Runtime Node application, on page 49
4. Install iWD Manager	<ul style="list-style-type: none"> Procedure: Creating the iWD Configuration database, on page 50. Procedure: Installing iWD Manager, on page 51.
5. Install iWD Rules, iWD Capture, and iWD Data Mart.	<ul style="list-style-type: none"> Procedure: Installing iWD Rules, on page 59. Procedure: Installing iWD Capture, on page 60. Procedure: Installing iWD Data Mart, on page 61. Procedure: Creating the iWD Data Mart Database, on page 62.
6. Create an iWD Tenant and iWD Solution in iWD Manager	Procedure: Creating an iWD Tenant and iWD Solution in iWD Manager , on page 65 .
7. Install and run the iWD Setup Utility to configure various mandatory configuration objects.	<ul style="list-style-type: none"> Procedure: Installing the iWD Setup Utility, on page 68 Procedure: Using the iWD Setup Utility, on page 69
8. Install iWD Stat Server Extensions.	Procedure: Installing the iWD Stat Extensions , on page 78 .
9. Link the iWD components with Genesys Configuration Server.	Procedure: Creating services in iWD Manager , on page 80 .
10. Configure logging for iWD Manager and iWD Runtime Node.	Procedure: Configuring logging for iWD Manager and iWD Runtime Node , on page 82
11. Update the Interaction Server databases and Event Log DAP.	Procedure: Updating the Interaction Server databases , on page 85 .

Task Summary: Installing iWD 8.0 (Continued)

Objective	Related Procedures and Actions
12. Configure remaining iWD services and objects.	See Chapter 4, “Configuration,” on page 97 .
13. Optional: Install and configure the sample application to learn more about the functionality of iWD and business rules.	See “Sample Application” on page 94 .

Preparing for Installation

This section describes how to prepare for the iWD installation. It contains the following topics:

- “[Configuration of Application Servers](#)”
- “[Installation Prerequisites](#)”
- “[CD Structure](#)” on [page 41](#)

Note: Stop your Tomcat web application server. Stopping is optional for WebSphere Application Server.

Configuration of Application Servers

It is necessary to configure application servers, both Apache Tomcat (Tomcat) and IBM WebSphere (WebSphere) in order to successfully run the iWD.

Installing on Tomcat

If Tomcat is running as Windows service:

- Add the following Java options to the Tomcat service configuration:
 - `-XX:MaxPermSize=128m`
 - `-Dcom.genesyslab.platform.commons.connection.factory.class=com.genesyslab.platform.commons.connection.impl.netty.NettyConnectionFactory`
- Configure the initial and maximum memory pools to 512 and 1024 megabytes (or 512 and 1500, if iWD Data Mart will be running on that particular instance of Tomcat).

If Tomcat is running as Windows console application:

- Add the following to the `setenv.bat` file:

```
set JAVA_OPTS=-XX:MaxPermSize=128m -Xms512M -Xmx1024M
-Dcom.genesyslab.platform.commons.connection.factory.class=com.
genesyslab.platform.commons.connection.impl.netty.NettyConnectionFa
ctory
```

- If this instance will be running iWD Data Mart, use `-Xmx1500M` or more instead of `-Xmx1024M`, if allowed by the operating system.

On UNIX machines:

- Edit the `setenv.sh` file and add the following:


```
export JAVA_OPTS="-XX:MaxPermSize=128m -Xms512M -Xmx1024M
-Dcom.genesyslab.platform.commons.connection.factory.class=com.
genesyslab.platform.commons.connection.impl.netty.NettyConnection
Factory"
```

 - If this instance will be running iWD Data Mart, use `-Xmx1500M` or more instead of `-Xmx1024M`, if allowed by the operating system.

`setenv.sh` and `setenv.bat` files:

By default, `setenv.sh` and `setenv.bat` files are not present after the installation of Tomcat, so you need to create them manually and correctly configure the access rights on UNIX machines properly for these files.

Installing on WebSphere

- Log in to the WebSphere Integrated Solution Console.
- Select `Servers>Application Servers>` and select a server from the list.
- Then continue to `>Server Infrastructure, Java and Process Management>Process Definition>Java Virtual Machine`.
- In `Generic JVM Arguments`, add these settings, if not present:
 - `-XX:MaxPermSize=128m`
 - `-Dcom.genesyslab.platform.commons.connection.factory.class=com. genesyslab.platform.commons.connection.impl.netty.NettyConnectionFactory`
- Configure the initial and maximum heap size, using the same guidelines as for Tomcat.
- If installing on WebSphere 7.0 or later, add this `Generic JVM` parameter:
 - `-Dorg.ajax4jsf.cache.CacheFactory=org.ajax4jsf.cache.OSCacheCacheFactory`
- Restart the application server.

Installation Prerequisites

Before you start the installation, ensure that the environment meets the following prerequisites:

- Interaction Server 8.0.1 or later is installed on a computer that can communicate with the computer on which the iWD node software is going to be installed. Refer to “Interaction Server Installation” on [page 42](#) for information about how to install an instance of Interaction Server to use with iWD.
- A supported web application server, such as Tomcat or WebSphere, is installed on the computer(s) on which iWD will be installed. Tomcat can be downloaded from the Apache website at <http://tomcat.apache.org/download-60.cgi>

Note: The user under whose context the iWD Runtime Node web application (for example, `iwd_node`) is running **must** have security permission to **write** to the local file system. Files are copied to the `iwd_node/WEB-INF/config` directory when an iWD solution is deployed.

- A supported database server is installed for the iWD Manager database and the iWD Data Mart.
- Java SDK is installed. For customers using Tomcat, Genesys only supports Java SDK 6. Java SDK can be downloaded from the following website: <http://java.sun.com/javase/downloads/index.jsp>

Note: Refer to the *Genesys Supported Operating Environment Reference Manual* for supported Java application servers and database servers.

- iWD 8.0 is compatible only with release 7.6.200.04 and higher of Genesys Universal Routing Server.

CD Structure

iWD software is packaged on the following CDs:

- iWD Runtime, Rules, and Manager CD
- iWD Data Mart, Stat Extensions, and Capture CD
- Optionally, you might use the iWD WebSphere MQ Capture Adapter, a component that allows iWD to capture tasks from source systems that use IBM WebSphere MQ as a message bus. This component is packaged on its

own CD. Refer to the *iWD WebSphere MQ Capture Adapter Reference Guide* for information about this component, its CD, and installation instructions.

Note: Beginning in version 8.0.2, eServices offers integrated **JMS** Capture Point functionality. Refer to the eServices documentation for more information. Version 8.0.210 of eServices (specifically, the Interaction Server component) also includes integrated XML File Capture Point functionality. Version 8.1.0 of eServices (the Interaction Server component) includes an integrated Database Capture Point.

iWD Runtime, Rules, and Manager CD

This CD contains the following components:

- **iWD Setup Utility**—A wizard-based utility that is used during the initial deployment of iWD to create many of the required configuration objects in Genesys Configuration Server.
- **iWD Runtime Node**—The core components of iWD that handle tasks after they are captured from a source system.
- **iWD Manager**—A graphical user interface (GUI) for both technical and business configuration of the iWD solution, including the configuration of business rules. iWD Manager is also used for real-time management of tasks.
- **iWD Rules**—The components of iWD that allow classification and prioritization rules to be executed against tasks.

iWD Data Mart, Stat Extensions, and Capture CD

This CD contains the following components:

- **iWD Capture**—Out-of-the-box Capture Adapters that allow iWD to capture tasks from a wide variety of source systems through Web Services, XML files, or direct database access.
- **iWD Stat Extensions**—Stat Server Java Extensions that provide Stat Server clients, such as CCPulse+, the ability to request and display current-day statistics from iWD's Data Mart.
- **iWD Data Mart**—A repository for iWD reporting data including current-day and historical statistics.

Interaction Server Installation

Interaction Server is required for iWD 8.0. If you are an existing eServices customer, and Interaction Server and its databases are already installed and

configured for your environment, you can proceed with [Procedure: Installing the iWD Runtime Node](#), on [page 46](#). Otherwise, perform the steps in the following procedure.

Procedure: Installing Interaction Server and its databases

Purpose: To install an instance of Interaction Server and create the Interaction Server database and Event Log database.

Note: The following procedure is a general procedure. Please work with your enterprise's database administrator to follow the specific procedure that is required by your database management system and your enterprise policies.

Prerequisites

- The Interaction Management CD.
- Review the requirements for technical licensing. See “Technical Licensing” on [page 21](#).

Start of procedure

Create the Databases

1. Ensure that the database server is running.
2. Log on to the database server's administrative interface (such as Oracle Enterprise Manager).
3. Create a new database (for example, `ixn_db`). This database will be used by Interaction Server to store interaction data.
4. Create a new database (for example, `ixn_event log_db`). This database will be used to store event-logging details (refer to the *eServices (Multimedia) 8.0 User's Guide* for more information about the Event Log database).

Create the DAP Application Objects

5. In Configuration Manager or Genesys Administrator, create two Database Access Point (DAP) Application objects: one for the Interaction Server database, and one for the Event Log database.
 - a. On the General tab, enter a name for the DAP and select the DB Server Application object that will be used by the DAP to connect to the database.

- b. On the DB Info tab, enter the details that will be used to connect to the database:
 - DBMS Name: The name of the host on which the DBMS is located.
 - DBMS Type: The type of database management system that is being used in your environment.
 - Database Name: The name that you specified for the database.
 - User Name: The user name of the DBMS user.
 - Password: The password that is used to connect to the DBMS.
 - Re-enter Password: Enter the password again.
- c. Add the following option to the Event Log Database Access Point (DAP), and also to the Interaction Server Interactions DAP application object.

On the Options tab, create a new section called jdbc. In the jdbc section, create an option called url. For the option value, enter your jdbc connection string to the Interaction Server Event Log database or Interaction Server Interactions database that is running on your database server.

If you are using Oracle database, use jdbc:evl:oracle:@ rather than jdbc:oracle:thin:@ or other providers.

Example: jdbc:evl:oracle:@//<oracle_host>:1521/<SID>.
- d. Only for the Interaction Server Event Log DAP:

On the Options tab, create a new section called logger-settings.
- e. In this section, create a new option called batch-size and set its value to 100.
- f. Click Apply to save the configuration.

Create the Interaction Server Application Object

6. Create the Interaction Server Application object. On the Interaction Management CD, in the templates folder, you will find the application template for Interaction Server. Import the template into Configuration Manager or Genesys Administrator
7. Create a new Application object, based on the template.
 - a. On the General tab, enter a name for your Interaction Server.
 - b. On the Server Info tab, select the host on which Interaction Server will be installed, and enter the port that Interaction Server will use to communicate.
 - c. On the Start Info tab, enter any text into the Working Directory, Command Line, and Command Line Arguments fields (these fields will be repopulated with the correct information when Interaction Server is installed, but something must be entered in these fields now in order to save the Application object).
 - d. On the Connections tab, add connections to the two DAPs that you created earlier in this procedure.

- e. The Options tab contains the configuration options for Interaction Server. Refer to the *eServices (Multimedia) 8.0 Reference Manual* for detailed descriptions of the options. Be sure to update the options for the license file.

There is a specific Interaction Server configuration option named `completed-queues` that specifies a list of queues for completed interactions. When an interaction is placed into one of these queues, the `CompletedAt` timestamp is set for that interaction. This is also the timestamp that will be used to calculate the Age of the interaction that is displayed on the Global Task List. Therefore, you should configure this option appropriately based on how you want this Age. For example, you may wish to set it to: `iWD_Completed`, `iWD_Canceled`, `iWD_Rejected`

Section: settings

Option name: `completed-queues`

Valid values: comma-separated list of queue names

Click **Apply** to save the configuration.

- 8. Open the properties of the Universal Routing Server (URS) Application object. On the Connections tab, add a connection to the Interaction Server that you just created.
- 9. A Multimedia Switching Office and Multimedia Switch must be created in Genesys Configuration Database, to support Stat Server and URS operations. Refer to the *eServices 8.0 Deployment Guide* for more details on these topics.

Install Interaction Server

- 10. Install Interaction Server.
 - a. From the computer on which Interaction Server will be installed, locate and double-click `Setup.exe` in the `InteractionServer` folder of the Interaction Management CD.
 - b. Click **Next** in the **Welcome** screen.
 - c. Select **Interaction Server** in the **Select Application Type** screen. Click **Next**.
 - d. Enter the login details to connect to Configuration Server and click **Next**.
 - e. Click **Next** in the **Client Side Port Configuration** screen (if you want to use client-side port configuration, refer to the *Genesys 8.0 Security Deployment Guide* for more information).
 - f. Select the **Interaction Server Application** that you configured in [Step 7](#) from the list and click **Next**.
 - g. Click **Install** to install Interaction Server.
 - h. Click **Finish** when the installation has been completed.

Initialize the Interaction Server Databases

11. Navigate to the installation directory for Interaction Server. On Windows, the default installation directory is `C:\Program Files\GCTI\eservices 8.0.1\Interaction Server`. Locate the script folder.
 - a. In the script folder, locate the folder for your DBMS type.
 - b. From your database administrative interface, run the `isdb_<Database Type>.sql` script on the Interaction Server database.
 - c. Run the `eldb_<Database Type>.sql` script on the Event Log database.

End of procedure

iWD Runtime Node Installation

This section describes the procedures that are used to install iWD Runtime Node. The iWD Runtime Node must be installed on a Java web application server before it can run iWD services. When deploying services for multiple tenants and solutions, each tenant and solution should have its own dedicated runtime node, so that the deployment of services to one runtime node will not affect the other runtime nodes that are running. Multiple runtime nodes can be set up to distribute load by running them on different physical computers. See “Solutions and Services” on [page 136](#) for more information about Solutions.

You might choose to have multiple Runtime Nodes under a single Solution in order to have a more distributed architecture, or for high availability. If you create additional Runtime Nodes on your application server, with corresponding Runtime Nodes created through iWD Manager, each of these Runtime Nodes in iWD Manager must map to a separate Runtime Node application in Configuration Server. Each of the Runtime Node Applications in Configuration Server must have a connection to the Interaction Server Application configured on its `Connections` tab.

Note: For information about deploying iWD Services to run on multiple Runtime Nodes, see “Deploying Services on Multiple Runtime Nodes” on [page 222](#).

Procedure: Installing the iWD Runtime Node

Purpose: To install an instance of the iWD Runtime Node.

Prerequisites

- The environment meets the requirements that are described in “Installation Prerequisites ” on [page 40](#).
- The web application server is stopped.
- The iWD Runtime, Rules, and Manager CD.

Start of procedure

1. Log into Configuration Manager or Genesys Administrator and import the iWD Runtime Node Application template from the iWD Runtime, Rules, and Manager CD.
2. Create a new Application object, based on the template. (Refer to the *Genesys Administrator Help* for specific information about creating new applications.)

Note: The iWD Runtime Node Application will show as a Third Party Server in Configuration Manager and Genesys Administrator.

3. On the General tab, enter a name for the iWD Runtime Node.
4. On the Server Info tab, enter the Host and Port that iWD Runtime will use.
5. On the Start Info tab, enter any text in the Working Directory, Command Line, and Command Line Arguments fields.

Note: The Application object cannot be saved if these fields are left blank, but because the actual values will not be used, you can enter any text.

6. On the Connections tab, add a connection to Interaction Server.
7. Click OK to save the Application object.
8. Locate and double-click `setup.exe` in the Runtime Node directory (`iwd_node`) of the iWD Runtime, Rules, and Manager CD.
9. The iWD Runtime Node Installation Wizard opens. Click Next in the Welcome screen.
10. Select the web container (for example, Apache Tomcat or WebSphere) and click Next.

11. If you selected WebSphere in [Step 10](#), select the appropriate JDK from the list and click Next.

Note: This is not the JDK which will be used by Websphere. This is necessary to properly configure the scripts which will be used to build the WAR archive.

12. In the Choose Destination Folder window, browse to the directory in which you want the iWD Runtime Node Java application part to be installed.
13. **If you selected Apache Tomcat in [Step 10](#)**, browse to the Home directory for your Apache Tomcat installation (for example, C:\Program Files\Apache\Tomcat60\). The Runtime Node components will be installed in the selected directory under the webapps folder.
If you selected WebSphere in [Step 10](#), browse to the arbitrary directory. iWD Runtime Node Java application part will be installed in that directory. After installation of iWD Runtime Node and other necessary iWD components you will need to build the WAR archive as described in “Post-Installation Steps When Using WebSphere Application Server” on [page 63](#) and install using WebSphere Integrated Solutions Console.
14. Click Next, then Install, then Finish.

End of procedure

Next Steps

- If you are using Java 6.0:
 - When iWD Runtime Node and iWD Manager have been installed, navigate to your web application server's \WEB-INF\lib folder and delete jaxb-api.jar and jaxb-impl.jar. These files should be deleted because their presence may lead to error messages due to compatibility issues between the versions of these .jar files and Java 6.0.

Note: Do **not** delete these files if you are using Java 5.0. Check your JAVA_HOME environment variable to verify which version is in use.

- If you are using the Statistics Adapter service in your environment, you must update the iWD Runtime Node application as described in [Procedure: Associating a Person's account with the iWD Runtime Node application](#).
- Install iWD Manager as described in “iWD Manager Installation” on [page 49](#).

Procedure:

Associating a Person's account with the iWD Runtime Node application

Purpose: This procedure is required only if you plan to deploy the Statistics Adapter service. This service is optional. Refer to “Statistics Adapter” on [page 170](#) for more information about this service. If deployed, the Statistics Adapter service will need to update the options in your Stat Server application, and it does this through the iWD Runtime Node application that has been configured in Configuration Server. For this reason, you need to ensure that your iWD Runtime Node is configured so that it can make the required changes to the Stat Server application on behalf of a user with appropriate security permissions.

Start of procedure

1. In Configuration Manager or Genesys Administrator, locate the Application for the iWD Runtime Node where you will be running the Statistics Adapter service.
2. Select the Security tab.
3. In the Log On As section, select This Account and click the browse button.
4. In the Add User dialog box, select a user who is a member of the Administrators Access Group and who has Full Control permissions for the iWD Runtime Node application.
5. Click OK to close the Add User dialog box.
6. Click OK to save and close the application's properties.

End of procedure

iWD Manager Installation

This section describes the procedures that are used to install iWD Manager. It contains the following topics:

- [iWD Configuration Database Preparation, page 50](#)
- [iWD Manager Installation, page 51](#)

iWD Configuration Database Preparation

The iWD Configuration database must be created prior to installation of iWD Manager.

Note: The iWD Configuration database is distinct from the Genesys Configuration database.

Procedure: Creating the iWD Configuration database

Purpose: The iWD Manager application utilizes a database in which all of the configuration data is stored. This database must be created before the iWD Manager application can be installed and accessed.

Note: The following procedure is a general procedure. Please work with your enterprise's database administrator to follow the specific procedure that is required by your database management system and your enterprise policies.

Start of procedure

1. Ensure that the database server is running.
2. Log on to the database server's administrative interface (such as Oracle Enterprise Manager).
3. Create a new database user account (for example, `iwdmanageruser`). This account will be used by iWD Manager to access the database.
4. Create a new database (for example, `iwdmanagerdb`). This database will be used by iWD Manager to store configuration data.
5. Ensure that the user who was created in [Step 3](#) has full access to the database, including the following permissions:
 - CREATE TABLE
 - CREATE INDEX
 - CREATE VIEW
 - CREATE TRIGGER
 - CREATE SEQUENCE

End of procedure

iWD Manager Installation

The iWD Manager application must be installed on a Java application server before it can be accessed. iWD Manager and iWD Runtime must be installed before any other iWD components can be installed.

Procedure: Installing iWD Manager

Purpose: To install the iWD Manager application. Installation of iWD Manager saves the required database scripts in the working directory. These scripts must be run against the iWD Configuration database and the Interaction Server database. This procedure describes how to install iWD Manager, as well as how to update the databases.

Prerequisites

- The web application server (such as Tomcat) is stopped.
- The iWD Configuration database has been created (see [Procedure: Creating the iWD Configuration database](#), on page 50).
- The iWD Configuration database is accessible.
- The computer on which the iWD Manager is going to be installed has network access to the computer that is hosting Genesys Configuration Server. Users of iWD Manager will be authenticated through Genesys Configuration Server.

Start of procedure

1. Log into Configuration Manager or Genesys Administrator and import the iWD Manager Application template from the iWD Runtime, Rules and Manager CD.
2. Create a new Application object based on the template. (Refer to the *Genesys Administrator Help* for specific information about creating new applications.) On the General tab, enter a name for your iWD Manager application.

Note: The iWD Manager Application will show as a Third Party Application in Configuration Manager and Genesys Administrator.

3. On the Connections tab, add a connection to Interaction Server.
4. Click Save to save the iWD Manager application.

5. Locate and double-click `setup.exe` in the iWD Manager directory of the iWD Runtime, Rules, and Manager CD.
6. The iWD Manager Installation Wizard opens. Click **Next** in the **Welcome** screen.
7. Select the web container (for example, Tomcat or WebSphere) and click **Next**.
8. If you selected WebSphere in [Step 7](#), select the appropriate JDK from the list.

Note: This is not the JDK which will be used by WebSphere. This is necessary to properly configure the scripts which will be used to build the WAR archive.

9. If you selected Apache Tomcat in [Step 7](#), browse to the Home directory for your Apache Tomcat installation. The iWD Manager components will be installed in the selected directory.
10. Click **Next**.
11. In the **Connection Parameters to the Configuration Server** screen, enter the login details to connect to Genesys Configuration Server and then click **Next**:

Host name: The host of Genesys Configuration Server
Port: The port that is used by Genesys Configuration Server
User name: The user name of the Person (or User) as defined in Genesys Configuration Manager or Genesys Administrator.
Password: The password that is associated with the Person (or User).
12. Choose the destination location for iWD Manager. If you selected Websphere in [Step 7](#), both supporting files and iWD Manager Java application part will be installed in that location. After installation of iWD Manager and other iWD components you will need to build the WAR archive as described in “Post-Installation Steps When Using WebSphere Application Server” on [page 63](#) and install using WebSphere Integrated Solutions Console.

You can accept the default or browse to another location on your computer. Click **Next**.
13. Select the database type that will be used by the iWD Configuration database, which was created in [Procedure: Creating the iWD Configuration database](#), on [page 50](#).
14. Enter the parameters that are used to connect to the iWD Configuration database in the next screen.

Enter the following information:

- **DB Server Host:** The name of the computer on which the database is located.
- **Database Name:** The name of the iWD Configuration database (as specified in [Procedure: Creating the iWD Configuration database](#), on [page 50](#)).
- **User Name:** The name of the user that is used to connect to the database.
- **Password:** The password that is used to connect to the database.

Click Next

15. Enter the host name and port of the computer on which the backup Genesys Configuration Server is running. If there is no backup Configuration Server in your environment, specify the primary Configuration Server host and port. Click Next.
16. In the Ready to Install screen, click Install to begin the installation of iWD Manager.
17. When installation has been completed, click Finish.
18. Optional step: Encode your database password. A file named `passwordEncoder.cmd` (or `passwordEncoder.sh` for UNIX-based operating systems) file is included when you install iWD Manager. This utility can be run to encode the database password that appears in the `ibd.properties` file, which is located in `<web application server directory>/webapps/ibd_manager/WEB-INF/classes` (the password is in plain text in the `ibd.properties` file by default).

Note: In order for the password encoder to work, the JDK/JRE `bin` directory must be added to the `PATH` system environment variable. For example, if the JDK is installed in `c:\Program Files\Java\jdk1.5.0_21\` then `c:\Program Files\Java\jdk1.5.0_21\bin\` should be in the `PATH` system environment variable.

- a. Open a Windows command-line window (go to Start->Run and enter `cmd` in the Run dialog box).
- b. Navigate to the directory in which the `passwordEncoder.cmd` file is located (for example, `cd C:\Program Files\GCTI\iWDManager\passwordEncoder` where the `cd` command is used to change the directory).
- c. Enter `passwordEncoder <unencoded password>` (for example, if the password is `genesys` you would type in `passwordEncoder genesys`).
- d. The command-line window will display the encoded version of the password

- e. In the `iwd.properties` file, replace the unencoded version of the password string with the encoded version (`iwd.configDatabase.password=`).
- f. Change the value of the `iwd.configDatabase.passwordEncoded` property to `true`.
- g. Save the `iwd.properties` file. Below are two sample files. The first shows an `iwd.properties` file before the password was encoded. The example shows the same file after the password was encoded.

Sample file with unencoded password

```
iwd.configDatabase.url=jdbc:sqlserver://iwd80vm;databaseName=iwd
managerdb
iwd.configDatabase.username=genesys
iwd.configDatabase.password=genesys
iwd.configDatabase.passwordEncoded=false
iwd.configDatabase.driverClassName=com.microsoft.sqlserver.jdbc.
SQLServerDriver
iwd.configDatabase.hibernateDialect=org.hibernate.dialect.SQLSer
verDialect
iwd.configDatabase.type=mssql

iwd.cfgServerHost=localhost
iwd.cfgServerPort=2020
iwd.cfgServerBackupHost=localhost
iwd.cfgServerBackupPort=2020

iwd.host=maestro_01
```

Sample file with encoded password

```
iwd.configDatabase.url=jdbc:sqlserver://iwd80vm;databaseName=iwd
managerdb
iwd.configDatabase.username=genesys
iwd.configDatabase.password=Z2VuZXN5cw==
iwd.configDatabase.passwordEncoded=true
iwd.configDatabase.driverClassName=com.microsoft.sqlserver.jdbc.
SQLServerDriver
iwd.configDatabase.hibernateDialect=org.hibernate.dialect.SQLSer
verDialect
iwd.configDatabase.type=mssql

iwd.cfgServerHost=localhost
iwd.cfgServerPort=2020
iwd.cfgServerBackupHost=localhost
iwd.cfgServerBackupPort=2020

iwd.host=maestro_01.
```

Note: You can use other Base64 encoders to encode your password as well. These can be found easily on the Web. One example is: <http://www.motobit.com/util/base64-decoder-encoder.asp>

End of procedure

Next Steps

- If you are using Java 6.0:
 - When iWD Runtime Node and iWD Manager have been installed, navigate to your web application server's \WEB-INF\lib folder and delete `jaxb-api.jar` and `jaxb-impl.jar`. These files should be deleted because their presence may lead to error messages due to compatibility issues between the versions of these `.jar` files and Java 6.0.

Note: Do **not** delete these files if you are using Java 5.

- Install the remaining iWD components. See “[Installing iWD Rules, iWD Capture, and iWD Data Mart](#)”.
- Run the iWD Setup Utility. See “[iWD Setup Utility](#)” on [page 67](#).

iWD Configuration Database Upgrade Utility (dbup)

iWD Manager includes a command line database initialization/upgrade utility. This utility is named `dbup.cmd` and is found in the `dbup` folder of the iWD

Manager installation directory. To initialize or upgrade your database in a semi-automated way (fully automated upgrade occurs upon first launch of iWD Manager), make a copy of the `dbup.<database type>.properties` file (found in the `dbup` folder), rename it to `dbup.properties` and modify it to specify your database connection parameters. Then, in a command prompt, enter the following command:

```
dbup.cmd -d
```

This utility can also be used to *update* the iWD Configuration database.

Note: If you execute `dbup.cmd` with no switch (such as `-d`), the console window will display information about the actual database query that will be made to the database, without actually executing the query. It will only execute the query when you type `dbup.cmd -d`. This will allow you to review the query before you execute it.

If you enter `dbup.cmd` without the `-d` switch, and a SQL query is not displayed, that means that your database is already up to date and there are no SQL queries that should be executed.

iWD Manager Database Auto-Update

iWD Manager can initialize and upgrade its database automatically. This is done by creating the database and configuring iWD Manager to use that database. No manual execution of database scripts is required.

At first launch iWD Manager will connect to the database and will allow the user to perform Create/Upgrade procedure from the login screen.

Enabling ADDP Connections Between iWD and the Genesys Suite

To enable ADDP in connections between various parts of iWD and the rest of Genesys suite, follow the steps below.

Enabling/Disabling ADDP for iWD Manager Application

Procedure:

Enabling/disabling ADDP for the iWD Manager application

Start of procedure

1. After installation, in the `iwd.properties` file, configure the following parameters:
 - `iwd.addpTimeout`
 - `iwd.addpRemoteTimeout`.By default ADDP is enabled and timeouts are set to 10 seconds.
2. To disable ADDP, set timeouts to 0 (zero).

End of procedure

Enabling/Disabling ADDP for iWD Runtime Node

Procedure:

Enabling/disabling ADDP for the iWD Runtime node

Start of procedure

1. In Genesys Administrator or Genesys Configuration Manager, open the iWD Runtime Node application.
2. On the **Connections** tab, enable ADDP for the connection to Interaction Server.
3. Set local and remote timeouts. Genesys recommends 10 seconds as normally sufficient.
4. In iWD Manager, under the solution which is deployed on a particular runtime node, configure the following parameters for Configuration Server Connector and Interaction Server Connector services:
 - `addpTimeout` (seconds)
 - `addpRemoteTimeout` (seconds).

Genesys recommends setting the timeouts to the same values as configured in Genesys Administrator or Configuration Manager for the Runtime Node application.

End of procedure

Enabling/Disabling ADDP for BCMS

Procedure:

Enabling/Disabling ADDP for Business Context Management Services

Start of procedure

1. In Genesys Administrator or Genesys Configuration Manager, open the BCMS application.
2. On the **Connections** tab, enable ADDP for the connection to BCMS. No additional configuration is required in the iWD Manager.

End of procedure

Installing iWD Rules, iWD Capture, and iWD Data Mart

When iWD Runtime Node and iWD Manager have been installed, the remaining iWD components can be installed. It does not matter the order in which you install the following components, but either iWD Runtime Node or iWD Manager **must** be installed first.

Note: At this stage in the installation process, you can also install the iWD WebSphere MQ Capture Adapter. Because the iWD WebSphere MQ Capture Adapter is a separate sellable product, its installation procedures are located in the *iWD 8.0 WebSphere MQ Capture Adapter Reference Guide*.

This section contains the following procedures:

- [Installing iWD Rules, page 59](#)
- [Installing iWD Capture, page 60](#)
- [Installing iWD Data Mart, page 61](#)

Note: Each of these components must be installed on the iWD Manager Host as well as the iWD Runtime Node host. The installation will prompt you to select whether you are installing on the iWD Manager host, the iWD Runtime Node host, or if you are installing to both (in other words, if iWD Manager and iWD Runtime Node are on the same host, select both).

When these components have been installed, run the iWD Setup Utility. See “iWD Setup Utility” on [page 67](#).

Note: When installing iWD components, stop your Tomcat web application server. Stopping is optional for WebSphere.

Procedure:

Installing iWD Rules

Purpose: To install an instance of iWD Rules for your iWD 8.0 Solution.

Prerequisites

- iWD Manager and/or iWD Runtime Node are installed.

Start of procedure

1. From the server that is running iWD Manager, navigate to the iWD Rules folder of the iWD Runtime, Rules, and Manager CD. Locate and double-click Setup.exe.
2. Click **Next** in the **Welcome** screen.
3. iWD components must be installed on the host that is running iWD Manager as well as the host that is running iWD Runtime Node. If both are installed on the same host, select iWD Manager and iWD Runtime Node. Otherwise, select only iWD Manager or iWD Runtime Node. Click **Next**.
4. Select the appropriate iWD Manager instance from the list that is displayed if it is installed on this host, and click **Next**. If you selected both iWD Manager and iWD Runtime Node, or only iWD Runtime Node, on the previous screen, you will be prompted to select the appropriate instance of iWD Runtime Node.
5. Choose a destination folder for iWD Rules. Either accept the default location or browse to a different location. Click **Next**.
6. Click **Install** to install iWD Rules. Click **Finish** when the installation has been completed.
7. If iWD Runtime Node is running on a different host, you will need to install iWD Rules on that host as well. Repeat steps [Steps 1](#) through [6](#) from the iWD Runtime Node host, selecting iWD Runtime Node when prompted.

End of procedure

Procedure: Installing iWD Capture

Purpose: To install an instance of iWD Capture for your iWD 8.0 Solution.

Prerequisites

- iWD Manager and/or iWD Runtime Node are installed.

Start of procedure

1. From the server that is running iWD Manager, navigate to the iWD Capture folder of the iWD Data Mart, Capture, and Stat Extensions CD. Locate and double-click `Setup.exe`.
2. Click **Next** in the **Welcome** screen.
3. Choose a destination folder for iWD Capture. Either accept the default location or browse to a different location. Click **Next**.
4. Click **Install** to install iWD Capture. Click **Finish** when the installation has been completed.
5. If iWD Runtime Node is running on a different host, you will need to install iWD Capture on that host as well. Repeat steps [Steps 1](#) through [4](#) from the iWD Runtime Node host, selecting iWD Runtime Node when prompted.

End of procedure

Notes:

1. For information about installing the iWD MQ WebSphere Capture Adapter, refer to the *iWD 8.0 WebSphere MQ Capture Adapter Reference Guide*.
2. Beginning in version 8.0.2, eServices offers integrated **JMS** Capture Point functionality. Refer to the eServices documentation for more information. Version 8.0.210 of eServices (specifically, the Interaction Server component) also includes integrated XML File Capture Point functionality. Version 8.1.0 of eServices (the Interaction Server component) includes an integrated Database Capture Point.

Procedure: Installing iWD Data Mart

Purpose: To install an instance of iWD Data Mart for your iWD 8.0 Solution.

Notes: Each iWD tenant solution requires its own Data Mart.

Genesys recommends putting the iWD Data Mart services on a separate Runtime Node than that which hosts the rest of the iWD services. This is because the iWD Data Mart services are CPU and memory intensive.

Prerequisites

- iWD Manager and/or iWD Runtime Node are installed.

Start of procedure

1. From the server that is running iWD Manager, navigate to the iWD Data Mart folder of the iWD Data Mart, Capture, and Stat Extensions CD. Locate and double-click `Setup.exe`.
2. Click **Next** in the **Welcome** screen.
3. Choose a destination folder for iWD Data Mart. Either accept the default location or browse to a different location. Click **Next**.
4. Click **Install** to install iWD Data Mart. Click **Finish** when the installation has been completed.
5. If iWD Runtime Node is running on a different host, you will need to install iWD Data Mart on that host as well. Repeat steps [Steps 1](#) through [4](#) from the iWD Runtime Node host, selecting iWD Runtime Node when prompted.

End of procedure

Next Steps

- Create the iWD Data Mart Database. See [Procedure: Creating the iWD Data Mart Database](#).
- If you will be using the Genesys Interactive Insights for iWD product for historical reporting based on the iWD Data Mart, you must enable several aggregates that are not enabled by default. Please see the “Reading iWD Aggregation” section of the *Interactive Insights 8.0 Deployment Guide*.

Procedure: Creating the iWD Data Mart Database

Purpose: To create the iWD Data Mart database.

Prerequisites

- iWD Data Mart is installed. See [Procedure: Installing iWD Data Mart](#), on [page 61](#).

Note: The following procedure is a general procedure. Please work with your enterprise's database administrator to follow the specific procedure that is required by your database management system and your enterprise policies.

Start of procedure

1. Ensure that the database server is running.
2. Log on to the database server's administrative interface (such as Oracle Enterprise Manager).
3. Create a new database (for example, `iwddatamart`). This database will be used by iWD Data Mart to store data.
4. Ensure that there is a user, who has access to the Data Mart database, who has the following permissions:
 - `CREATE TABLE`
 - `CREATE INDEX`
 - `CREATE VIEW`
 - `CREATE TRIGGER` (Oracle)
 - `CREATE SEQUENCE` (Oracle)
5. The iWD Data Mart database will be initialized automatically the first time the Database Service and Kettle ETL Service are started. See "Database Service" on [page 169](#) for more information about the Database Service. If the Database Service's `AutoSynchronize` option is selected, this initialization is automatic, and the Database Service will also check for updates to the iWD Data Mart database whenever a new version of iWD Data Mart is installed. The `AutoSynchronize` option will also initialize ETL plug-ins.

If your database administrator prefers to inspect the SQL statements prior and to execute the database initialization and/or upgrade manually, iWD provides a command-line utility for this called `dbup_dm.cmd`. See "[iWD Data Mart Database Upgrade Utility \(dbup_dm\)](#)" for more information about this utility.

End of procedure

iWD Data Mart Database Upgrade Utility (dbup_dm)

iWD Data Mart includes a command line database initialization/upgrade utility. This utility is called `dbup_dm.cmd` and is found in the `dbup` folder of the iWD Data Mart installation directory.

To initialize or upgrade your database, rename the `dbup.template.properties` file (found in the `dbup` folder) to `dbup.properties` and modify it to specify your database connection parameters. Then, in a command prompt, enter the following command:

```
dbup_dm.cmd -d
```

This utility can also be used to *update* the iWD Data Mart database.

Notes: If you execute `dbup_dm.cmd` without any switch (such as `-d`), the console window will display information about the actual database query that will be made to the database, without actually executing the query. Only when you type `dbup_dm.cmd -d` will it actually execute the query. This will allow you to review the query before you execute it.

If you type `dbup_dm.cmd` without the `-d` switch, and you do not see any SQL query displayed, then you know that your database is already up to date and there are no SQL queries that should be executed.

Post-Installation Steps When Using WebSphere Application Server

After the installation of iWD Manager, iWD Runtime Node and all the components, it is necessary to build the WAR archives and install them into WebSphere using Integrated Solutions Console.

Procedure:

Building WAR archives for iWD Manager and iWD Runtime Node

Start of procedure

1. Browse to the directory which was specified during installation of iWD Manager and continue to subdirectory `\webapps`.
2. Launch the `iWD_Manager.bat` file. This will create the `iwd_manager.war` file
3. Browse to the directory which was specified during installation of iWD Runtime Node and continue to subdirectory `\webapps`.

4. Launch the `iWD_RuntimeNode.bat` file. This will create the `iwd_node.war` file
5. Log in to Websphere Integrated Solutions Console.
6. Uninstall existing iWD Manager and iWD Node applications, if they are present.
7. Install iWD applications, and select the prepared WAR files when prompted.
8. When installation is completed, adjust the order of classloaders for each installed iWD application. By default, classloader order is Parent first, then Application. iWD requires the order to be Application first, then Parent.
9. To change the order of the classloaders, in WebSphere Integrated Solutions Console, click on Application, click Manage Modules, click on Module (one per application), then change the classloader order to Application, then Parent.
10. Click Save.
11. Start the application.

End of procedure

Importing iWD Configuration XMLs

Procedure: **Import default configuration XMLs into iWD Manager**

Prerequisites

iWD Manager and iWD Node as well as all required components are installed. Application servers are started (if running on WebSphere – applications are started).

Start of procedure

1. Open this link in the browser: `http://app_server_host:port/iwd_manager/`
2. Log in.
3. If iWD Manager configuration database is not initialized using dbup or provided SQL scripts, perform Create/Upgrade Configuration Database from login screen
4. Log into iWD Manager.
5. In iWD Manager, open Import/Export in the General section of the System tenant.

6. Import the following files:

- `<iWD Manager installation directory>\config\iwd.xml.`
- If you are going to use XML message transformation:
`<iWD Manager installation directory>\config\iwd_transform.xml`

Note: The XML files are saved to the installation directory when iWD application or component is installed.

- Capture:
`<iWD Capture installation directory> \config\iwd_capture.xml.`
 - Data Mart:
`<iWD Datamart installation directory>\config\iwd_datamart.xml`
and
`<iWD Datamart installation directory>\config\iwd_reporting.xml.`
 - Rules:
`<iWD Rules installation directory>\config\iwd_rules.xml.`
7. After each import a message appears at the bottom of the screen to indicate whether the import was successful.

End of procedure

Creating the Tenant and Solution

After the IPs are installed, you must create an iWD tenant and iWD solution in iWD Manager to proceed with the configuration of iWD.

Procedure:

Creating an iWD Tenant and iWD Solution in iWD Manager

Purpose: To create a Tenant and Solution in iWD Manager for your iWD installation.

Note: It is recommended that you do **not** create any Solutions and Services under the System Tenant. You should do so under a Managed Tenant. The recommended role for the System Tenant is to be the place where you create all of your Modules and Components (which is done by importing the various configuration xml files); these Modules can then be inherited by the Managed Tenants and assigned or unassigned as needed at the Managed Tenant level.

Start of procedure

1. Login to iWD Manager:
 - a. Open a web browser and enter
`http://<servername>:8080/iwd_manager/`
 where <servername> server is the name of the server on which iWD Manager is installed and 8080 is the application server port.
 - b. Log into iWD Manager by using a user name and password of a user that is configured in Genesys Configuration Server. If you are connecting to a Genesys Configuration Server that was recently installed, there will be a default account with user name = default and password = password. For CME Application Name, enter the name that you gave your iWD Manager Application object in [Step 2 of Procedure: Installing iWD Manager](#), on [page 51](#).
2. Select General from the navigation panel and select the System tenant from the Tenant Selection drop-down box (see “iWD Manager Overview” on [page 97](#) for more information about the iWD Manager user interface).
3. In the navigation tree, select New under Managed Tenants.
4. Create a new iWD Tenant that will correspond to the Tenant in Configuration Server. Enter a name and description for the iWD Tenant and select the corresponding Genesys Configuration Server Tenant from the CME Tenant drop-down box. Refer to “Managed Tenant Details” on [page 118](#) for more information about iWD Tenants. Remember that tenants in iWD correlate one-to-one with tenants in Configuration Server.

Create the iWD Tenant

Note: If you have a single-tenant Genesys Configuration Server, the CME Tenant will be Resources. In a multi-tenant configuration, you will have to select the appropriate Tenant from the list.

5. Save the new iWD tenant.
6. Select the tenant created in [Step 4](#) from the Tenant Selection drop-down box.
7. Select Services from the navigational panel.
8. In the navigation tree, select New Solution Instance.
9. On the Solution Instances screen in iWD Manager, enter a name and description for the new solution (see “Solution Details” on [page 137](#) for information about the various properties that can be configured for an iWD Solution). Click Save.

Create the iWD Solution

Note: Make note of the Solution ID that is assigned to the newly created solution. This ID will be required when you run the iWD Setup Utility (the ID will still be visible in iWD Manager as well).

Create the Runtime Node

10. The new solution will appear in the navigation tree (if you do not see it, verify your tenant is selected in the tenant drop-down box, and select the Services navigation section). Under your solution in the navigation tree, select `Runtime Nodes`.

Note: Runtime nodes configured here point to the URLs of iWD Runtime Node java applications, which are containers for the services.

11. On the `Runtime Nodes` panel, click `New` to create a new Runtime Node (see “Runtime Nodes” on [page 138](#) for information about the various properties that can be configured for a runtime node)
12. Select the Runtime Node Configuration Management Environment application from the list.

Note: This action links runtime node Java applications to the application objects in the Genesys Configuration Server. Each iWD service running within a container (iWD Runtime Node java application) is authenticated by Configuration Server on behalf of the Configuration Management Environment application that is selected while configuring the runtime node for the solution.

13. Click `Save`.

End of procedure**Next Steps**

- Install and run the iWD Setup Utility. See “[iWD Setup Utility](#)”.
- Create as many iWD solutions as are required for your business needs. Each iWD solution requires its own dedicated Stat Server. For each iWD solution that you create, run the iWD Setup Utility and install the Stat Server Java Extension.

iWD Setup Utility

Installing and running the iWD Setup Utility is the next step in installing iWD 8.0. The iWD Setup utility:

- Imports business processes. The iWD Setup Utility includes the following business processes:
 - iWDBP—the default business process, explained in detail in Appendix B, “iWD Business Process (IWDBP),” on [page 251](#).

- Standard Genesys to IWD adapter—the business process used to insert into iWDBP to serve Genesys standard open media channels, explained in Appendix C, “Adapting the iWD Business Process for Standard Genesys Channels,” on [page 275](#). Creates the Agent Group that is used in the business process.
- ABC IWD Simple BP—the business process used to insert into existing business processes, explained in Appendix D, “iWD Rules and Existing Business Processes,” on [page 307](#).
- Create an Agent Group called IWD in your Genesys Configuration Database. This is the name of the Agent Group that is used in the example Distribution routing strategy that is included in the standard iWD Business Process (IWDBP). Although all customers are expected to modify this Distribution routing strategy for their own needs, having the IWD Agent Group created out of the box will make it easier to use IWDBP to process interactions for testing purposes.
- Creates custom Interaction Server attributes. These attributes are visible in Configuration Manager under <tenant name> > Business Attributes > Interaction Custom Properties.

Note: You can create additional custom properties to map to any custom attributes used in your tasks. For more information about creating custom properties, refer to the “Interaction Properties” section of the *eServices (Multimedia) 8.0 User’s Guide*.

- Creates the capacity rule that includes the media type work item and provides the option to assign it to the tenant.
- Allows you to select and configure the Stat Server to use with iWD Stat Extensions.
- Adds the DAP for reporting in Interaction Server.
- Creates the connector objects for the iWD solution.

Procedure: Installing the iWD Setup Utility

Purpose: To install the iWD Setup Utility.

Start of procedure

1. On the iWD Runtime, Rules, and Manager CD find and double-click `setup.exe` in the `iwdSetupUtil` folder.
2. Click Next in the Welcome screen.
3. Browse to the desired destination folder, or accept the default. Click Next.

4. Click **Install**.
5. Click **Finish** on the **Installation Complete** screen.

End of procedure

Next Steps

- Use the iWD Setup Utility to prepare your environment for iWD deployment.

Procedure: Using the iWD Setup Utility

Purpose: To run the iWD Setup Utility to configure the objects required in Configuration Manager for your iWD installation.

Notes: The iWD Setup Utility can be run multiple times, as it is possible to have multiple iWD Solutions in your environment. The following procedure outlines the steps taken the first time the utility is run. Subsequent runs of the utility may result in some screens not being displayed, as the information has already been configured. Therefore, when you run the iWD Setup Utility it might not follow the exact procedure outlined below.

The iWD Setup Utility will **not** over-write the iWD business process if it has already been imported by a previous run of the iWD Setup Utility. So, if any customization has been made on the business process, running the iWD Setup Utility will not impact your customization, provided all strategy names are the same.

If you have made changes to the iWD business process, but would like to see the business process that is included in the iWD Setup Utility, you must export the customized business process from IRD, delete it, and then run the iWD Setup Utility again. Or, you can run the iWD Setup Utility against a different Tenant.

Prerequisites

- The iWD Setup Utility is installed. See [Procedure: Installing the iWD Setup Utility](#).
- iWD Manager and iWD Runtime Node have been installed. See [Procedure: Installing iWD Manager](#), on [page 51](#) and [Procedure: Installing the iWD Runtime Node](#), on [page 46](#).
- The iWD Data Mart has been installed and the iWD Data Mart database has been created. See [Procedure: Installing iWD Data Mart](#), on [page 61](#) and [Procedure: Creating the iWD Data Mart Database](#), on [page 62](#).

- Interaction Server 8.0.1 or later has been installed. If you are not an existing eServices customer, you will need to install an instance of Interaction Server to use with iWD refer to “Interaction Server Databases” on [page 85](#).

Start of procedure

1. Start the iWD Setup Utility. Go to Start > Programs > Genesys Solutions > iWD > iWD Setup Utility > Start iWD Setup Utility.
2. Enter login details to connect to the Genesys Configuration Server on the Login screen (shown in [Figure 12](#)), and click OK.

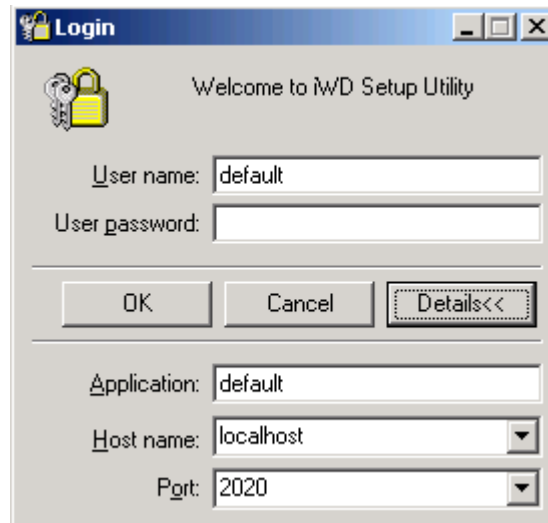


Figure 12: iWD Setup Utility Login screen

3. The iWD Setup Utility’s Welcome screen is displayed. Click Next.
4. If you are in a multi-tenant environment, you will be prompted to select the Configuration Server Tenant for which this iWD installation applies.

5. The Samples Placement screen (see [Figure 13](#)) is displayed. Browse to the desired destination directory for strategy files and click Next.

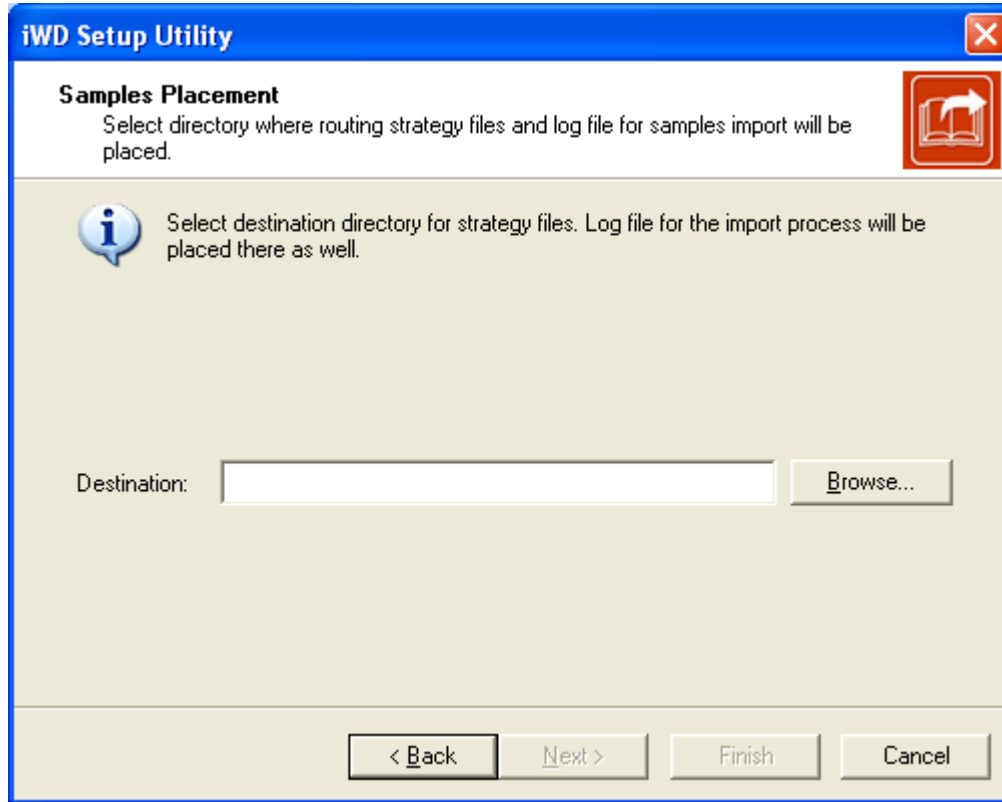


Figure 13: iWD Setup Utility Samples Placement screen

6. The iWD Setup Utility imports the configuration objects and relocates the strategy files. On the Import Completed screen, click Next.
7. The Resource Capacity Rules Selection screen is displayed (see [Figure 14](#)). To select the capacity rule, select the check box and click Next.

Note: Selecting this check box will create a capacity rule that can be used immediately, because any new Capture Point services that you create in iWD Manager will have a media type set to `workitem` by default. You are not required to use `workitem` as the media type. You can create new media types in Configuration Manager or Genesys Administrator (as Business Attribute objects) and you can use the new media types in any Capture Point services you create, as well as in any capacity rules you configure.

For more information about Business Attributes, refer to the following documents:

- *Universal Routing 8.0 Business Process User's Guide*
- *Universal Routing 8.0 Reference Manual*
- *eServices (Multimedia) 8.0 User's Guide*

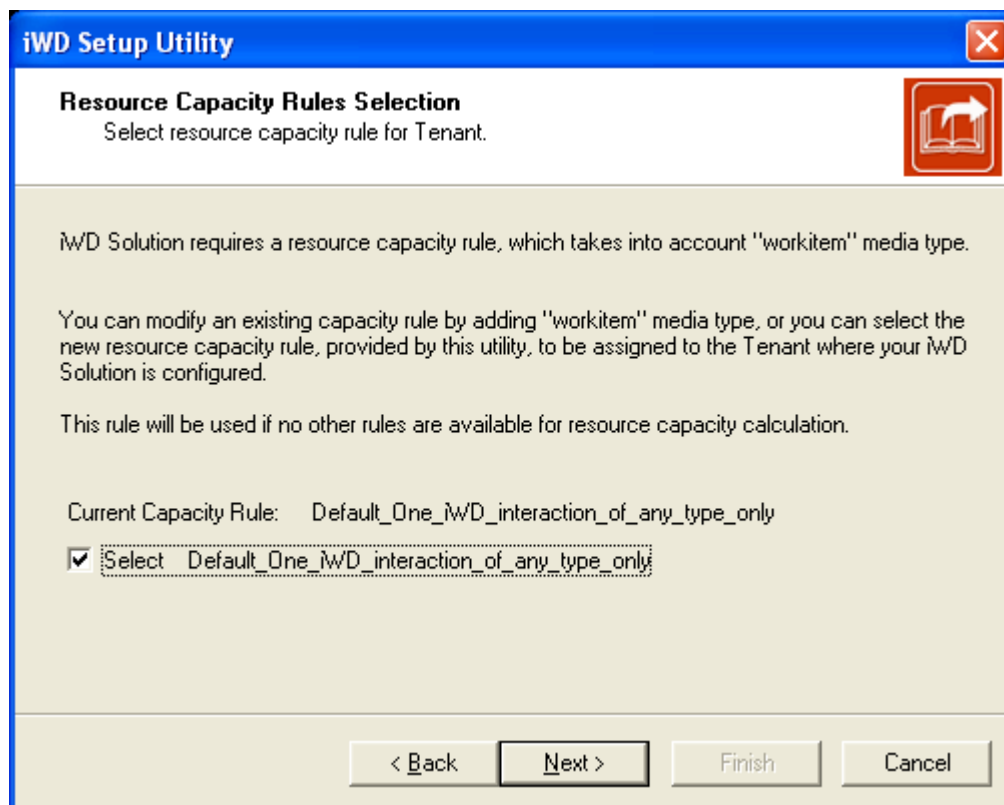


Figure 14: iWD Setup Utility Resource Rules Selection screen

Note: The iWD Setup Utility Resource Rules Selection screen (shown in [Figure 14](#)) mentions that a capacity rule is required that takes into account the `workitem` media type. This is not accurate, since iWD may use media types other than `workitem`. However, the default capacity rule that will be created by the iWD Setup Utility will be based on the media type `workitem`, as described in the Note on [page 71](#).

8. On the Interaction Server Selection screen (see [Figure 15](#)), browse to the Interaction Server you plan to use for this iWD Solution. Click Next.

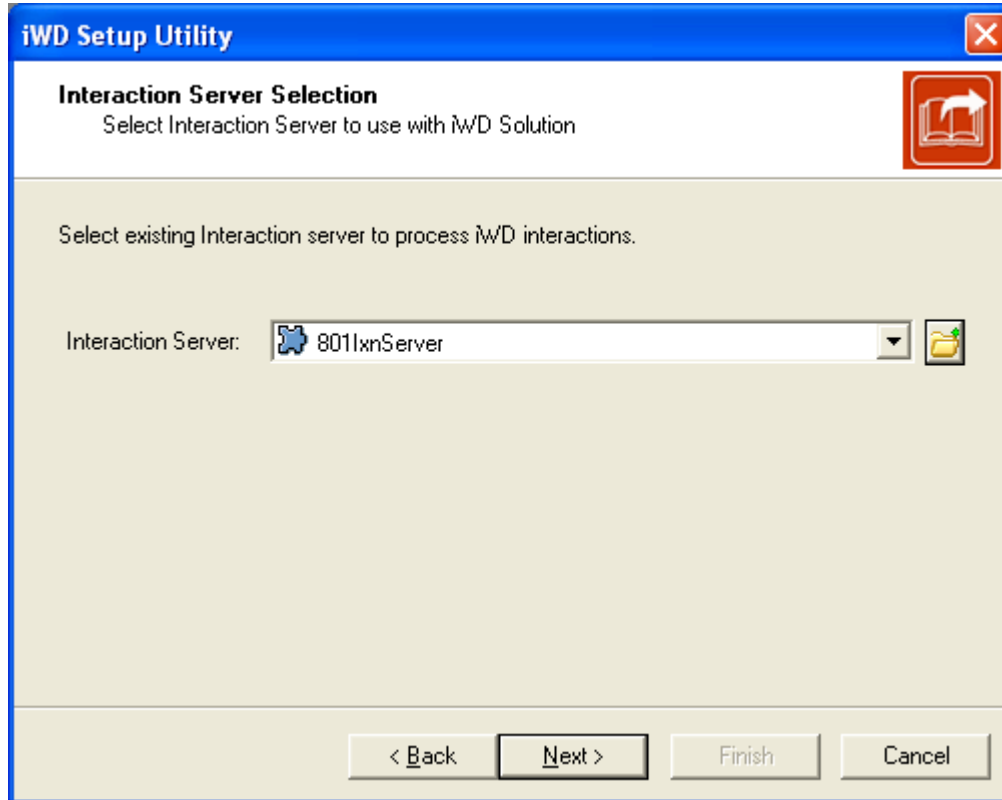


Figure 15: Interaction Server Selection

9. On the Database Access Point Selection screen, browse to the Database Access Point that is configured for the Interaction Server Event Log database, or create a new one if necessary. Click Next.

Note: This screen will not be displayed if the connections were added previously, as described in [Procedure: Installing Interaction Server and its databases](#), on [page 43](#).

10. On the Stat Server Selection screen, select the Stat Server to be used by iWD Stat Extensions for this solution. Click Next.

Note: Each iWD Solution requires its own dedicated Stat Server. For more information about installing and configuring Stat Server, refer to the *Framework 8.0 Stat Server Deployment Guide*.

11. On the iWD Data Mart Database Parameters screen, enter the following information and then click Next:

Host: Enter the host of the iWD Data Mart database server.

Port: Enter the port of the database server (default is 1433).

DBMS Type: Select the DBMS type from the drop-down list.

Database Name: Enter the name of the iWD Data Mart database (or the SID if using an Oracle database).

Note: The database name entered here must **exactly** match the name that was used when the iWD Data Mart database was created. iWD Stat Extensions reads data from the `gtl_stat` table in the iWD Data Mart database.

User Name: Enter the user name to connect to the database.

Password: Enter the password to connect to the database.

12. On the Virtual Queue Selection screen, browse to, or create, the Virtual Queue object on the Multimedia Switch to use for iWD Stat Extensions reporting. Only one virtual queue is needed for all Stat Server reporting. Refer to the *eServices (Multimedia) 8.0 Deployment Guide* and *eServices (Multimedia) 8.0 User's Guide* for more information about Multimedia Switches. Click Next.

13. On the iWD Business Context Management Service Selection screen (see [Figure 16](#)), enter the following information to create a new iWD Business Context Management Service to support Extended Service Protocol (ESP) requests from Interaction Server:

Application Name: Enter a name for the iWD Business Context Management Service.

Host: Select the host on which the service will be installed. Specify the host on which iWD Runtime Node is installed.

Port: Enter the port for the service.

Click Next.

This will create a new application in the Genesys Configuration database, which will be used when Interaction Server needs to make requests to the iWD rules engine to evaluate business rules.

Notes: Each Solution in iWD must have its own Business Context Management Service.

Refer to [page 221](#) for information about configuring multiple Business Context Management Services.

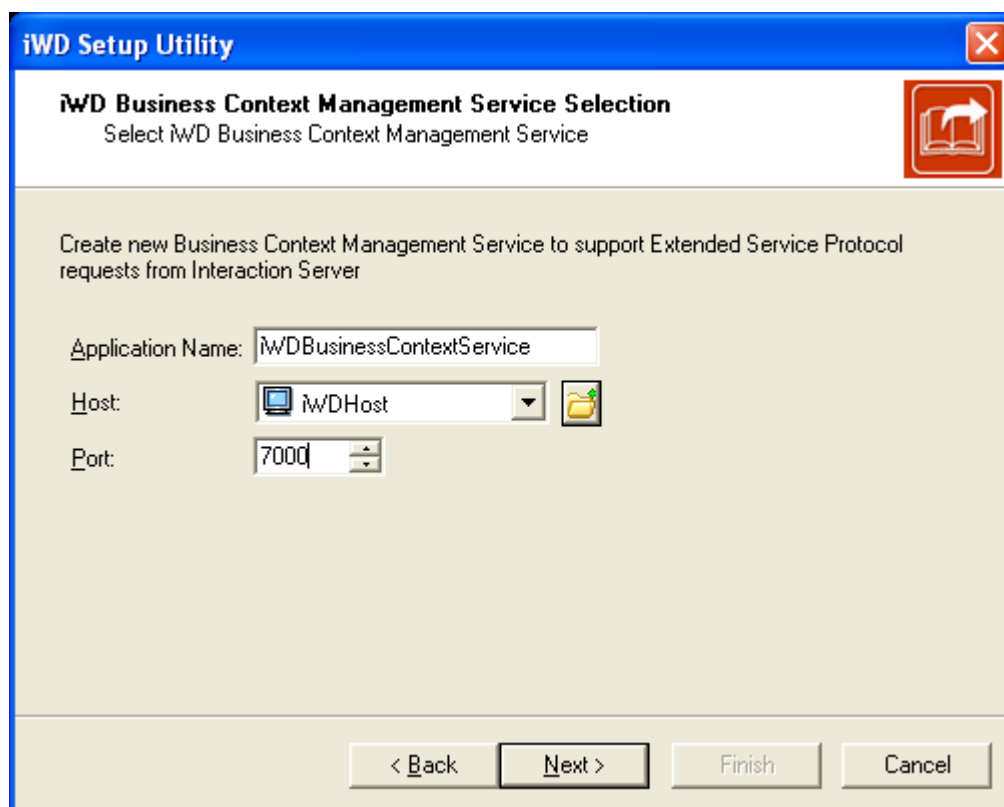


Figure 16: iWD Business Context Management Service

14. On the next screen (see [Figure 17](#)), enter the following information to create an iWD Runtime Node application. If you have manually created the runtime node application(s) in Configuration Manager, you can leave these fields empty and click Next to skip the creation of the runtime node application. This application's primary function will be to establish a

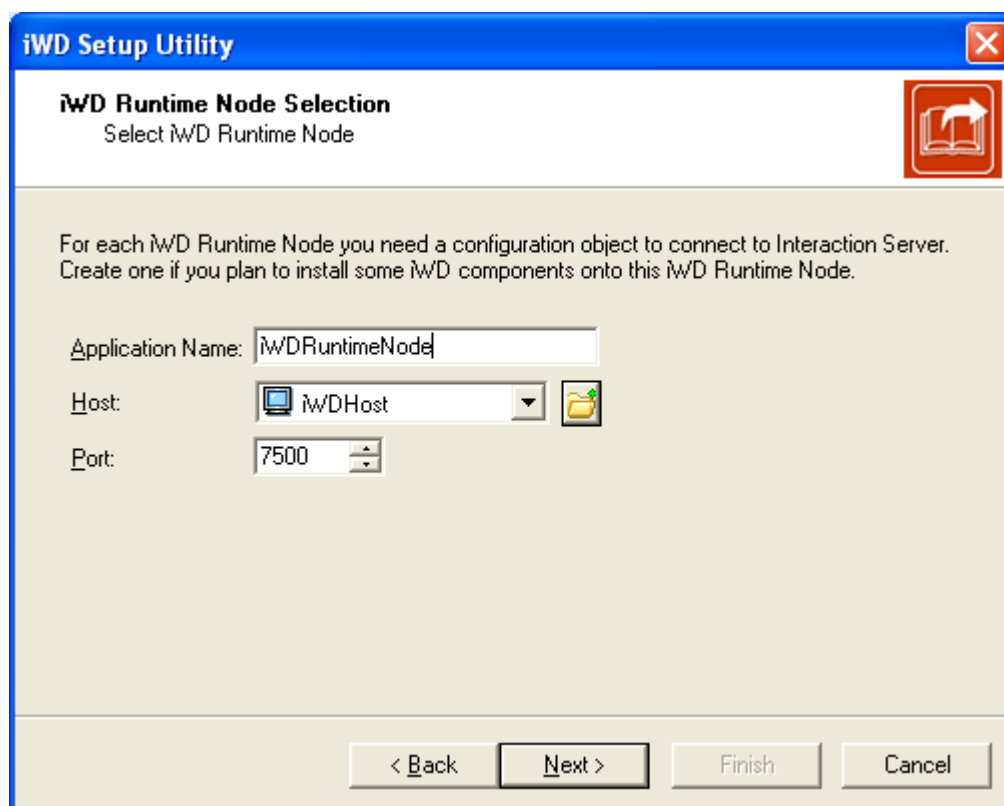
connection to Interaction Server on behalf of an actual iWD Runtime Node running on your application server:

Application Name: Enter a name for the iWD Runtime Node.

Host: Select the host on which the iWD Runtime Node application will be installed.

Port: Enter the port for the iWD Runtime Node application.

Click Next.



The image shows a Windows-style dialog box titled "iWD Setup Utility" with a subtitle "iWD Runtime Node Selection". The subtitle is followed by the instruction "Select iWD Runtime Node". Below this, a paragraph explains: "For each iWD Runtime Node you need a configuration object to connect to Interaction Server. Create one if you plan to install some iWD components onto this iWD Runtime Node." The form contains three input fields: "Application Name:" with the text "iWDRuntimeNode" entered; "Host:" with a dropdown menu showing "iWDHost" and a folder icon to its right; and "Port:" with a numeric spinner set to "7500". At the bottom, there are four buttons: "< Back", "Next >", "Finish", and "Cancel". The "Next >" button is highlighted with a black border.

Figure 17: iWD Runtime Node

15. The iWD Solution Selection screen is displayed. Enter the Solution ID for the solution that you created in [Procedure: Creating an iWD Tenant and iWD Solution in iWD Manager](#), on page 65.

The ESP server field is disabled, but it displays the name of the Business Context Management Service that you created in [Step 13](#) (see [Figure 18](#)). Click Next.

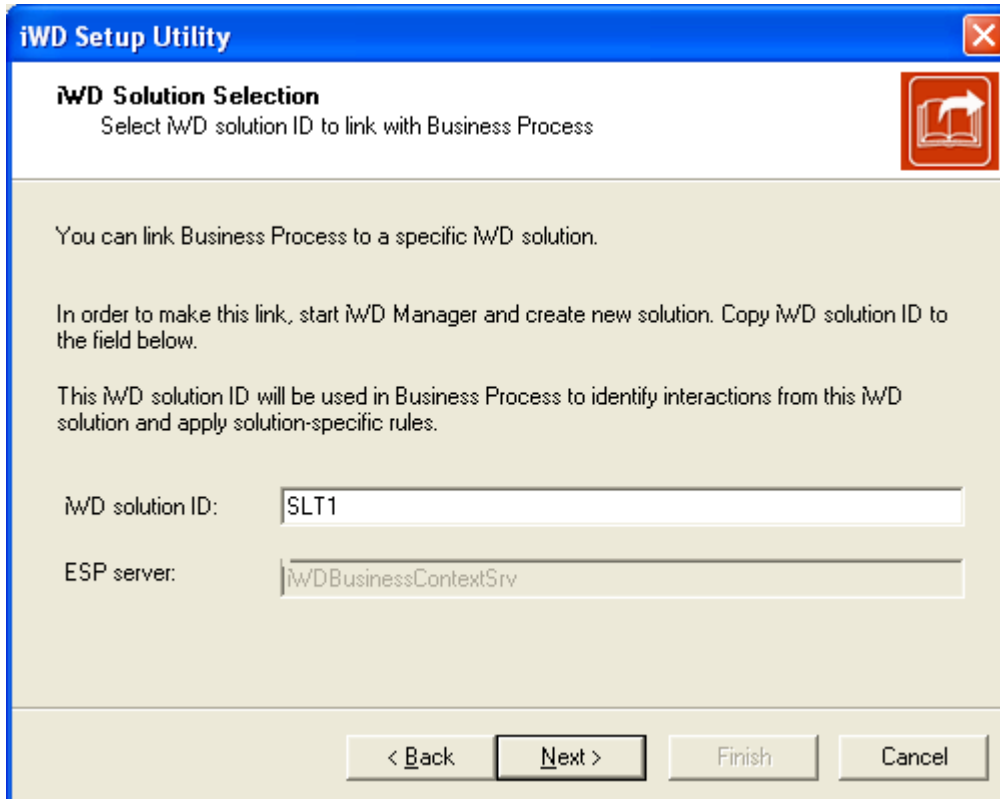
The image shows a screenshot of the 'iWD Setup Utility' window. The title bar is blue with the text 'iWD Setup Utility' and a close button. The main window has a light beige background. At the top, there's a section titled 'iWD Solution Selection' with a subtitle 'Select iWD solution ID to link with Business Process'. Below this, there's a paragraph: 'You can link Business Process to a specific iWD solution. In order to make this link, start iWD Manager and create new solution. Copy iWD solution ID to the field below. This iWD solution ID will be used in Business Process to identify interactions from this iWD solution and apply solution-specific rules.' There are two input fields: 'iWD solution ID:' with the value 'SLT1' and 'ESP server:' with the value 'iWDBusinessContextSrv'. At the bottom, there are four buttons: '< Back', 'Next >', 'Finish', and 'Cancel'. The 'Next >' button is highlighted with a black border.

Figure 18: iWD Solution Selection

16. Click Finish to complete the iWD Setup Utility.

End of procedure

Next Steps

- Install iWD Stat Extensions.

Installation of iWD Stat Extensions

After the iWD Setup Utility has completed, you can install the iWD Stat Extensions.

Procedure: Installing the iWD Stat Extensions

Purpose: To install the extensions needed to ensure Genesys Stat Server will work with your iWD 8.0 solution.

Prerequisites

- An instance of Stat Server is installed, dedicated for use with iWD. Refer to the *Framework 8.0 Stat Server Deployment Guide* for more information.

Start of procedure

1. From the server that is running Stat Server, navigate to the iWD Stat Extensions folder of the iWD Data Mart, Capture, and Stat Extensions CD. Locate and double-click Setup.exe.
2. Click Next on the Welcome screen.
3. Select the appropriate Stat Server instance from the list that is displayed and click Next.
4. Click Install to install iWD Stat Extensions. Click Finish when the installation has been completed.

End of procedure

Stat Server Configuration Options

The required Stat Server configuration options will already be configured if you have used the iWD Setup Utility and installed iWD Stat Extensions. Each option is described briefly below.

java-extensions section

During installation, a new option is added to the java-extensions. The option BPR_iWD_Extension.jar is added with a value of true.

java-extensions-bpr-iwd section

The java-extensions-bpr-iwd section contains options which specify the JDBC connection driver and parameters for access to the iWD Data Mart database.

The only option whose value shouldn't be changed under normal circumstances is java-extension-jar. The value of this option is the name of the BPR iWD extension jar-file.

The rest of options are described briefly below:

- `jdbc-driver`: The class name for the corresponding JDBC driver. Valid values include:
 - `com.inet.tds.TdsDriver` (for MS SQL)
 - `com.inet.ora.OraDriver` (for Oracle)
 - `com.ibm.db2.jcc.DB2Driver` (for IBM DB2)
 - `com.mysql.jdbc.Driver` (for MySQL)
- `jdbc-url`: The JDBC URL, which describes RDBMS-specific access parameters. Below are some sample values:
 - `jdbc:inetdae7:hostname:1433?database=databasename` (for MS SQL)
 - `jdbc:inetora:hostname:1521:databasename` (for Oracle, if you are using the instance name of the database)
 - `jdbc:inetora:hostname:1521?service=<Service ID>` (for Oracle, if you are using the network service name of the database)
 - `jdbc:db2://hostname:50000/databasename` (for IBM DB2)
 - `jdbc:mysql://hostname:3306/databasename` (for MySQL)
- `user`: The user name for database access.
- `password`: The password for database access.
- `verbose`: The level to control debug information, provided in the Stat Server log file. Possible values are debug, trace, or standard.
- `refresh-interval`: The interval (in minutes) for data updates from database.

**Report Statistics
for each
Dimension on its
own Virtual Queue**

The iWD Stat Server Java Extension can be configured to report statistics in two different ways: all statistics for all dimensions can be reported on one Virtual Queue, or each dimension can have its statistics reported on its own Virtual Queue.

The iWD Setup Utility configures Stat Server to use the iWD Stat Server Java Extension to report all statistics on one Virtual Queue. If you want to change this, you have to modify the configuration manually.

To report each dimension on its own Virtual Queue:

- Set the the option “`dimension-mapping-1`” to the value `Virtual Queue`.
- Instead of using the Virtual Queue name as a value for the option `virtual-queue-name-1`, indicate the prefix that will be used for Virtual Queue names. For example, if `virtual-queue-name-1=dim-`, then Virtual Queues with the names `dim-CNT_T2_C106`, `dim-CNT_T2_C107` and so on, have to be created.

Link with Genesys Configuration Server

The next step in deploying your iWD solution is to link iWD with Genesys Configuration Server. Some configuration needs to be done in iWD Manager to connect the iWD solution with the Genesys configuration.

Procedure:

Creating services in iWD Manager

Purpose: To create the Logging Service, Configuration Server Connector, Interaction Server Connector, Rules, and iWD Business Context Management Services.

Note: For a complete list of iWD Services, listed in order of recommended configuration, refer to “iWD Services” on [page 146](#). This procedure only describes a few of the required services. Depending on your business needs, others will be required.

Start of procedure

1. Login to iWD Manager:
 - a. Open a web browser and enter `http://<servername>:port/iwd_manager/` where <servername> server is the name of the server on which iWD Manager is installed and port is the application server port.
 - b. Log into iWD Manager by using a user name and password of a user configured in Genesys Configuration Server. If you are connecting to a Genesys Configuration Server that was recently installed, there will be a default account with user name = default and password = password. For CME Application Name, enter the name you gave your iWD Manager Application object in [Step 2 of Procedure: Installing iWD Manager](#), on [page 51](#).
2. Select Services from the navigation panel and select your tenant from the Tenant Selection drop-down box.
3. Locate your solution in the navigation tree.
4. Create a Logging Service for the solution.
 - a. Select Services > New Service under the solution.
 - b. Enter a name for the Service, such as LoggingService.
 - c. Select Logging Service from the Service Template drop-down box.
 - d. Enter a description for the service.
 - e. Configure values for the rest of the properties. Refer to “Logging Service” on [page 152](#) for information about the properties and possible values.
 - f. Click Save to save the new service.
5. Create a Configuration Server Connector Service for the solution.
 - a. Select Services > New Service under the solution.
 - b. Enter a name for the Service, such as CfgServerConnector.

- c. Select Configuration Server Connector from the Service Template drop-down box.
 - d. Enter a description for the service.
 - e. Configure values for the rest of the properties. Refer to “Configuration Server Connector Service” on [page 153](#) for information about the properties and possible values.
 - f. Click Save to save the new service.
6. Create an Interaction Server Connector Service.
 - a. Select Services > New Service under the solution.
 - b. Enter a name for the service.
 - c. Select Interaction Server Connector from the Service Template drop-down box.
 - d. Enter a description for the service.
 - e. Configure values for the rest of the properties. Refer to “Interaction Server Connector Service” on [page 154](#) for information about the properties and possible values.
 - f. Click Save to save the new service.
7. Create a Rules Service.
 - a. Select Services > New Service under the solution.
 - b. Enter a name for the service.
 - c. Select Rules Service from the Service Template drop-down box.
 - d. Enter a description for the service.
 - e. Configure values for the rest of the properties. Refer to “Rules Service” on [page 167](#) for information about the properties and possible values.
 - f. Click Save to save the new service.
8. Create a Business Context Management Service.
 - a. Select Services > New Service under the solution.
 - b. Enter a name for the service.
 - c. Select Business Context Management Service from the Service Template drop-down box.
 - d. Enter a description for the service.
 - e. Configure values for the rest of the properties. Refer to “Business Context Management Service” on [page 168](#) for information about the properties and possible values. For the CMEApplicationName property, be sure to select the name of the Business Context Management Service that you created while using the iWD Setup Utility.

- f. Click Save to save the new service.

Note: Refer to [page 221](#) for information about configuring multiple Business Context Management Services.

9. Deploy the changes made to your Solution.

End of procedure

Next Steps

- Create as many iWD Solutions as required for your business needs. If you plan to configure reporting through CCPulse+, each iWD Solution will require its own dedicated Stat Server. For each iWD Solution you create, run the iWD Setup Utility and install the Stat Server Java Extension.
- Configure logging for iWD Manager and iWD Runtime Node. See “[Logging](#)”.
- Update the Interaction Server databases. See “Interaction Server Databases” on [page 85](#).

Logging

iWD Manager and iWD Runtime Node support creation of their own log files for troubleshooting purposes. As well, they both support centralized logging through Genesys Message Server. The parameters needed for both types of logging are configured in the `log4j.properties` file for each application. The following procedure explains how to configure the required parameters.

Procedure: **Configuring logging for iWD Manager and iWD Runtime Node**

Purpose: To configure logging for iWD Manager and iWD Runtime Node.

Prerequisites

- iWD Manager and iWD Runtime are both installed.
- For centralized logging, Genesys Message Server is installed. Refer to the *Framework 8.0 Management Layer User's Guide* for more information about Message Server.

Start of procedure

1. Find the `log4j.properties` file in the web application server's (Tomcat or WebSphere) `webapps` directory and open it in a text editor.

Note: For WebSphere, the application is located under
`WAS_root/profiles/<profile>/installedApps/<node, cell>/<application>.war.ear/<application>.war/WEB-INF/classes`

Example:

`/usr/IBM/WebSphere/AppServer/profiles/AppSrv01/installedApps/rs6000Node01Cell/iwd_manager_war.ear/iwd_manager.war/WEB-INF/classes`

2. The exact path is `webapps\iwd_node\WEB-INF\classes\log4j.properties` for iWD Runtime Node and `webapps\iwd_manager\WEB-INF\classes\log4j.properties` for iWD Manager.

Note: Be sure to remove the comment symbol (#) when you update the parameters.

3. For centralized logging, update the very first row of the `log4j.properties` file to include `centralized_node` at the end:
`log4j.rootLogger=INFO, runtime, centralized_node`
4. For centralized logging, update the following parameters with the Message Server host and Message Server port, respectively:
`log4j.appender.centralized_manager.MessageServerHost=[ToBeChanged:MSG_SRV_HOST]`
`log4j.appender.centralized_manager.MessageServerPort=[ToBeChanged:MSG_SRV_PORT]`
5. Update the following line to include the name of your iWD Manager or iWD Runtime application (this is the application name that you must use to set up alarms in Solution Control Server):
`log4j.appender.centralized_node.MessageServerClientName=iWD Runtime Node`
6. Configure the parameter to specify the level of logging to send to Message Server:

`log4j.appender.centralized_node.Threshold=INFO`

Threshold can be set to any of the following values:

- TRACE
- DEBUG
- INFO

- WARN
 - ERROR.
7. For information about various settings for the root logger, refer to log4j framework docs, such as <http://logging.apache.org/log4j/1.2/manual.html>.
 8. You can change the directory where the logs will be written by changing the value of this line:

```
log4j.appender.runtime.File=C:/GCTI/iWD/iwd_runtime.log
```

Note: It is strongly recommended that you only set the file path to a directory on a local machine, not a remote location such as a shared network drive. Logging to a remote location can severely impact performance

9. You can change the logging level by changing this line (sample is from an iWD Runtime log4j.properties file):

```
log4j.appender.runtime.Threshold=INFO
```

Possible values are Off, Warning, Error, Debug, Info, and Trace. See “Service Log Levels” on [page 152](#) for a description of each log level.

10. You can change the maximum file size of the logs by changing this line:

```
log4j.appender.runtime.MaxFileSize=256MB
```

11. Save your changes.

The following is a sample iWD Runtime log4j.properties file configured for logging:

```
# info, rolling 256MB each (2 GB max)
```

```
log4j.rootLogger=INFO, runtime, centralized_node
```

```
log4j.category.org.apache.myfaces.renderkit.html.util=ERROR
```

```
log4j.category.org.hibernate.util.JDBCExceptionReporter=FATAL
```

```
log4j.category.org.hibernate.event.def.AbstractFlushingEventListener=FATAL
```

```
log4j.category.org.apache.commons.httpclient=ERROR
```

```
log4j.category.org.apache.commons.digester=ERROR
```

```
log4j.category.org.codehaus.xfire.transport.http.HttpChannel=FATAL
```

```
log4j.category.org.codehaus.xfire.handler.DefaultFaultHandler=ERROR
```

```
log4j.appender.runtime=org.apache.log4j.RollingFileAppender
```

```
log4j.appender.runtime.Threshold=INFO
```

```
log4j.appender.runtime.File=C:/GCTI/iWD/iwd_runtime.log
```

```
log4j.appender.runtime.layout=org.apache.log4j.PatternLayout
```

```

log4j.appender.runtime.layout.ConversionPattern=%d{yyyy-MM-dd
HH:mm:ss} |%t| %p |%c| %m%n
log4j.appender.runtime.MaxBackupIndex=7
log4j.appender.runtime.MaxFileSize=256MB

# if you enable centralized_node appender then make sure you
change rootLogger to:
# log4j.rootLogger=INFO, runtime, centralized_node

log4j.appender.centralized_node=com.genesyslab.iwd.log.Centraliz
edAppender
log4j.appender.centralized_node.Threshold=INFO
log4j.appender.centralized_node.layout=org.apache.log4j.PatternL
ayout
log4j.appender.centralized_node.layout.ConversionPattern=%m
log4j.appender.centralized_manager.MessageServerHost=MsgSrvrHost
log4j.appender.centralized_manager.MessageServerPort=4050
log4j.appender.centralized_node.MessageServerClientName=iWD
Runtime Node

```

End of procedure

Interaction Server Databases

Once all iWD components have been installed, database update scripts have to be run on the Interaction Server databases and options need to be imported into your Event Log DAP.

Procedure: Updating the Interaction Server databases

Purpose: To run the update scripts on the Interaction Server databases for compatibility with iWD 8.0, import configuration options for the Event Log DAP, and configure the completed-queues option for Interaction Server.

Prerequisites

- Interaction Server is installed as described in [Procedure: Installing Interaction Server and its databases](#), on [page 43](#) (also refer to the *eServices (Multimedia) 8.0 Deployment Guide*).

- The installation of iWD components as outlined in this chapter is completed up to this point. In particular, iWD Manager must be installed. During iWD Manager installation, the database scripts that are required to update the Interaction Server databases are saved to the iWD Manager installation directory.

Note: The procedure below is a general procedure. Please work with your enterprise's database administrator to follow the specific procedure required by your database management system and your enterprise policies.

Start of procedure

1. Locate the `eventlog_update.sql` script in the `<iWD Manager directory>/sql_scripts/<Database Type>` directory and run it on the Interaction Server Event Log database.
2. Locate the `iwd_custom_fields.sql` script in the `<iWD Manager directory>/sql_scripts/<Database Type>` directory and run it on the Interaction Server database.
3. In Configuration Manager or Genesys Administrator, open the properties for your Interaction Server Application. In the `settings` section, locate the `completed-queues` option. This option specifies a list of queues for completed interactions. The `completed_at` timestamp in the Interaction Server database is set for an interaction when it is placed in one of these queues. Add `iWD_Completed` as a value. This is the name of the queue used for completed iWD interactions (the name of this queue is not configurable. `iWD_Completed` must be used). Save the changes.
4. In Configuration Manager or Genesys Administrator, open the properties for your Event Logger DAP Application. On the `Properties` tab, import `eventlog_options.cfg` to update the DAP's properties. The `eventlog_options.cfg` file can be found in the `config` directory after iWD Manager is installed.

End of procedure

Next Steps

Now you should start configuring the mandatory and optional services for your Solution. Jump ahead to “iWD Services” on [page 146](#) for information about the available services, their recommended order of configuration, and the properties that can be configured for each service. If you would like to review how to work with iWD Manager first, start at the beginning of [Chapter 4](#).

Installing iWD on UNIX-Based Operating Systems

The following iWD 8.0 Installation Packages (IPs) are supported on UNIX operating systems.

- iWD Manager
- iWD Runtime Node
- iWD Rules
- iWD Capture
- iWD Data Mart
- iWD Stat Extensions
- iWD WebSphere MQ Capture Adapter (optional)

Note: The iWD Setup Utility cannot be deployed on UNIX platforms; it can only be deployed on Windows hosts.

For the supported UNIX flavors and versions, please consult the *Genesys Supported Operating Environment Reference Manual*.

Configuring and installing iWD 8.0 on UNIX hosts follows the general procedure outlined earlier in this chapter.

Below are examples of what it might look like when you run the setup for each IP manually on a UNIX host. In these examples, the installation is being done on a Solaris host, and the iWD 8.0 version being installed is 8.0.001.04. You might have a slightly different experience if you are installing on a different UNIX host, or are installing a different version of iWD 8.0.

Note: If you are using Tomcat, please take into account the following:

- If the CATALINA_HOME environment variable is defined, installation will not ask where to install the Java application parts of the iWD applications and components; these will be installed to `$CATALINA_HOME/webapps/<iwd_manager, iwd_node/`
 - If this is not a desired behavior, please temporarily remove the CATALINA_HOME environment variable while installing iWD.
-

iWD Manager IP

```
Sun Microsystems Inc.   SunOS 5.10       Generic January 2005
bash-3.00# id
uid=0(root) gid=0(root)
bash-3.00# locale
LANG=en_US.UTF-8
```

```

LC_CTYPE="en_US.UTF-8"
LC_NUMERIC="en_US.UTF-8"
LC_TIME="en_US.UTF-8"
LC_COLLATE="en_US.UTF-8"
LC_MONETARY="en_US.UTF-8"
LC_MESSAGES="en_US.UTF-8"
LC_ALL=
bash-3.00# java -version
java version "1.5.0_20"
Java (TM) 2 Runtime Environment, Standard Edition (build
1.5.0_20-b02)
Java HotSpot(TM) Server VM (build 1.5.0_20-b02, mixed mode)
bash-3.00# echo $JAVA_HOME
/usr/jdk/instances/jdk1.5.0/jre
bash-3.00# echo $PATH
/usr/jdk/instances/jdk1.5.0/jre:/usr/bin
bash-3.00# ls
iwdCapAdaptMQS  iwdDataMart      iwdRules          iwdSetupUtil
iwdCapture      iwdManager       iwdRuntimeNode    iwdStatExt
bash-3.00# cd iwdManager
bash-3.00# ls
8000104
bash-3.00# cd 8*
bash-3.00# ls
ip
bash-3.00# cd ip
bash-3.00# ls
data.tar.gz          install.sh          read_me.html
genesys_silent.ini   ip_description.xml  tar
gunzip               iscript.tar.gz
tar_gunzip_license.txt
installer.tar.gz     ospatchlist.txt
bash-3.00# ./install.sh
*****
* Welcome to the Genesys 8.0 Installation Script *
*****

```

Installing iWD Manager, version 8.0.001.04

Please select your servlet container type by number:

1. Tomcat
 2. WebSphere
- =>1

Please enter the full path to your
Tomcat installation =>/var/apache/tomcat55

Please specify the type of used Database Server:

- 1) MS SQL Server
- 2) MySQL Server
- 3) Oracle Server


```
=>1
```

```
Please enter the Database Server hostname or IP address
```

```
=>135.225.54.208
```

```
Please enter the Database name =>iwdmanagerdb
```

```
Please enter the Database Server user name =>iwdmanageruser
```

```
Please specify the Database Server user password =>
```

```
Please enter the Configuration Server Host Name =>135.225.54.208
```

```
Please enter the Configuration Server Port =>2020
```

```
Please enter the Configuration Server Backup Host Name
```

```
=>135.225.54.208
```

```
Please enter the Configuration Server Backup Port =>2020
```

```
Please enter full path of the destination directory for installation
```

```
=>/tmp/iwd/iwdManager8000104
```

```
Extracting tarfile: data.tar.gz to directory:
```

```
/tmp/iwd/iwdManager8000104
```

```
acme/
```

```
acme/readme.txt
```

```
acme/web/
```

```
acme/web/confirmOrderForm.php
```

```
.....
```

```
.....
```

```
webapp/help/task_management.htm
```

```
webapp/help/Hide.gif
```

```
webapp/META-INF/
```

```
webapp/META-INF/MANIFEST.MF
```

```
Installation of iWD Manager, version 8.0.001.04 has completed  
successfully
```

```
bash-3.00#
```

iWD Runtime Node IP

```
bash-3.00# pwd
```

```
/tmp/iWD_IP/iwdRuntimeNode/8000104/ip
```

```
bash-3.00#
```

```
bash-3.00# ls
```

```
data.tar.gz
```

```
installer.tar.gz
```

```
iscript.tar.gz
```

```
tar
```

```

genesys_silent.ini      install.sh      ospatchlist.txt
tar_gunzip_license.txt
gunzip                  ip_description.xml  read_me.html
bash-3.00# ./install.sh
*****
* Welcome to the Genesys 8.0 Installation Script *
*****

Installing iWD Runtime Node, version 8.0.001.04

Please select your servlet container type by number:
1. Tomcat
2. WebSphere
=>1
Please enter the full path to your
Tomcat installation =>/var/apache/tomcat55

Please enter full path of the destination directory for installation
=>/tmp/iwd/iwdRuntimeNode8000104

Extracting tarfile: data.tar.gz to directory:
/tmp/iwd/iwdRuntimeNode8000104
.....
.....
webapp/ui/lib/codepress/license.txt
webapp/ui/lib/codepress/codepress.html
webapp/ui/lib/codepress/index.html
webapp/ui/lib/codepress/codepress.js
webapp/ui/lib/codepress/images/
webapp/ui/lib/codepress/images/line-numbers.png

Installation of iWD Runtime Node, version 8.0.001.04 has completed
successfully.

bash-3.00#

```

iWD Rules IP

```

bash-3.00# pwd
/tmp/iWD_IP/iwdRules/8000104/ip
bash-3.00# ls
data.tar.gz      installer.tar.gz  iscript.tar.gz
tar
genesys_silent.ini  install.sh      ospatchlist.txt
tar_gunzip_license.txt
gunzip            ip_description.xml  read_me.html
bash-3.00# ./install.sh

```

```

*****
* Welcome to the Genesys 8.0 Installation Script *
*****

Installing iWD Rules, version 8.0.001.04

Please select your servlet container type by number:
1. Tomcat
2. WebSphere
=>1
Please enter the full path to your
Tomcat installation =>/var/apache/tomcat55

Please enter full path of the destination directory for installation
=>/tmp/iwd/iwdRules8000104

Extracting tarfile: data.tar.gz to directory:
/tmp/iwd/iwdRules8000104
config/
config/iwd_rules.xml
iwd_rules.jar
lib/
lib/janino.jar
lib/drools-compiler.jar
lib/commons-jci-core.jar
lib/commons-jci-janino.jar
lib/drools-core.jar
lib/stringtemplate.jar

Installation of iWD Rules, version 8.0.001.04 has completed
successfully.

bash-3.00#

```

iWD Capture IP

```

bash-3.00# pwd
/tmp/iWD_IP/iwdCapture/8000104/ip
bash-3.00# ls
data.tar.gz          installer.tar.gz      iscript.tar.gz
tar
genesys_silent.ini   install.sh            ospatchlist.txt
tar_gunzip_license.txt
gunzip               ip_description.xml    read_me.html
bash-3.00# ./install.sh
*****
* Welcome to the Genesys 8.0 Installation Script *
*****

```

Installing iWD Capture, version 8.0.001.04

Please select your servlet container type by number:

1. Tomcat
2. WebSphere

=>1

Please enter the full path to your

Tomcat installation =>/var/apache/tomcat55

Please enter full path of the destination directory for installation

=>/tmp/iwd/iwdCapture8000104

Extracting tarfile: data.tar.gz to directory:

/tmp/iwd/iwdCapture8000104

config/

config/iwd_capture.xml

iwd_capture.jar

Installation of iWD Capture, version 8.0.001.04 has completed successfully.

bash-3.00#

iWD Data Mart IP

bash-3.00# pwd

/tmp/iWD_IP/iwdDataMart/8000104/ip

bash-3.00# ls

data.tar.gz installer.tar.gz iscript.tar.gz
tar

genesys_silent.ini install.sh ospatchlist.txt

tar_gunzip_license.txt

gunzip ip_description.xml read_me.html

bash-3.00# ./install.sh

```
*****
* Welcome to the Genesys 8.0 Installation Script *
*****
```

Installing iWD Data Mart, version 8.0.001.04

Please select your servlet container type by number:

1. Tomcat
2. WebSphere

=>1

Please enter the full path to your

Tomcat installation =>/var/apache/tomcat55

```
Please enter full path of the destination directory for installation
=>/tmp/iwd/iwdDataMart8000104
```

```
Extracting tarfile: data.tar.gz to directory:
/tmp/iwd/iwdDataMart8000104
config/
config/iwd_datamart.xml
config/iwd_reporting.xml
dbup/
dbup/iwd_common.jar
dbup/dbup.template.properties
dbup/dbup.oracle.properties
dbup/iwd_dm.jar
dbup/iwd_db_migrate.jar
dbup/readme.txt
dbup/dbup_dm
dbup/sqljdbc.jar
dbup/dbup.mssql.properties
dbup/dbup.mysql.properties
etl/
etl/kettle_mysql.properties
etl/aggregate_stats.kjb
etl/aggregate_intraday.kjb
.....
.....
etl/plugins/classif/aggregate_historical_aggregate.ktr
iwd_dm.jar
lib/
lib/edtftpj.jar
lib/javadb.jar
lib/jug.jar
lib/iwd_common.jar
lib/libformula.jar
lib/CacheDB.jar
lib/sqljdbc.jar
lib/kettle.jar
lib/jsch.jar
lib/jackcess.jar
sql_scripts/
sql_scripts/mysql/
sql_scripts/mysql/iwd_dm_mysql.sql
sql_scripts/oracle/
sql_scripts/oracle/iwd_dm_oracle.sql
sql_scripts/mssql/
sql_scripts/mssql/iwd_dm_mssql.sql
```

```
Installation of iWD Data Mart, version 8.0.001.04 has completed
successfully.
```

```
bash-3.00#
```

iWD Stat Extensions IP

```

bash-3.00# pwd
/tmp/iWD_IP/iwdStatExt/8000009/ip
bash-3.00# ls
data.tar.gz          installer.tar.gz      iscript.tar.gz
tar
genesys_silent.ini    install.sh            ospatchlist.txt
tar_gunzip_license.txt
gunzip                ip_description.xml    read_me.html
bash-3.00# ./install.sh
*****
* Welcome to the Genesys 8.0 Installation Script *
*****

Installing iWD Stat Extensions, version 8.0.000.09

Please enter full path of the destination directory for installation
=>/tmp/iwd/iwdStatExt8000009

Extracting tarfile: data.tar.gz to directory:
/tmp/iwd/iwdStatExt8000009
java/
java/ext/
java/ext/BPR_iWD_Extension.jar
java/lib/
java/lib/sqlj.zip
java/lib/mysql-connector-java-3.0.17-ga-bin.jar
java/lib/bprDbServices.jar
java/lib/db2jcc.jar
java/lib/0pta2000.jar
java/lib/Seropto.jar

Installation of iWD Stat Extensions, version 8.0.000.09 has
completed successfully.

bash-3.00#

```

Sample Application

iWD 8.0 includes a sample application (ACME) that can be used to experiment with iWD. The sample application is included on the iWD Manager CD. When iWD Manager is installed, a subfolder called `acme` is created in the iWD Manager installation directory. A `readme.txt` file is saved to this folder and contains information about the ACME sample application.

Note: The ACME sample application requires MySQL.

More information about how to set up and use the ACME sample can be found at the following location:

<https://sites.google.com/a/iwdlab.com/iwd8/general/acme-sample-application>



Chapter

4

Configuration

This chapter describes how to configure various objects used in intelligent Workload Distribution. This chapter contains the following sections:

- [iWD Manager Overview, page 97](#)
- [General Configuration, page 111](#)
- [Modules and Components, page 125](#)
- [Solutions and Services, page 136](#)
- [iWD Services, page 146](#)
- [Departments and Processes, page 182](#)

iWD Manager Overview

iWD Manager is a thin-client web application for configuring and managing intelligent Workload Distribution. It can be accessed via a web browser. For a complete list of supported web browsers, refer to the *Genesys Supported Operating Environment Reference Manual*. The application URL is installation-specific (see “iWD Manager Installation” on [page 49](#) of [Chapter 3](#)). The following is an example of a typical iWD Manager URL:

`http://<host>:<port>/iwd_manager/`

Note: Browser cookies must be enabled for the iWD Manager application to function correctly.

General Conditions for Configuring an iWD Manager User

- To access the Global Task List, the Person or User must be associated with a Place in Genesys Configuration. If the Person or User is not an Agent, the Place can be configured as an option on the Annex tab of the

Person/User object. Create a section `iWD` (if it does not already exist) and within the `iWD` section, create an option `iWDManagerPlace` with the value of a valid `Place` name.

- If you want a user to have access to a specific managed tenant in `iWD`, that user must have at least `Read` access to the `Configuration Server` tenant that is linked to that `iWD` managed tenant—either directly or by being a member of an `Access Group` that does.
- The user must have at least `Read & Execute` permissions to the `iWD Manager` application in `Configuration Server`, and `Read` permissions to the `Configuration Server Host` object where the `Interaction Server` is running, either directly or by being a member of an `Access Group` that does.
- If the user is going to have access to the `Global Task List`, then the user must have at least `Read` permissions to these application objects in `Configuration Server`:
 - `Interaction Server`
 - The `Database Access Point` for the `Interaction Server` database
 - The `Database Access Point` for the `Interaction Server Event Log` database.

These permissions may be applied directly to the user, or by the user being a member of an `Access Group` that has such permissions.

- The `Place` that must be associated with the user must be a `Place` configured under the `Configuration Server` tenant that maps to the `iWD` managed tenant. (This presumes that the `Interaction Server` application also has an association with this `Configuration Server` tenant.)
- The `Place` is only used to give the user access to the `Global Task List`. If the user only needs to log into `iWD Manager` to perform other functions but does not need access to the `Global Task List`, they need not have a `Place` configured.
- If you want to restrict what the user can do in `iWD Manager`, you must configure a `Security Role` in `iWD Manager` that maps to an `Access Group` to which the user belongs.
- The user should not have any access to the `Environment` in `Configuration Server` unless they are an `Administrative` user who should have full permissions to `iWD Manager`. If you provide even `Read` access to the `Environment`, it will override any security policy you have configured for them in `iWD Manager`.
- If the user is created under a `Configuration Server` child tenant, then the following additional conditions must be met:
 - If you want the user to have access to both the `iWD SYSTEM` tenant as well as the `iWD` managed tenant to which the `Configuration Server` tenant maps, then this user must be a member of at least two `Access Groups`: one under the `Environment` tenant and one under the child tenant. In both cases, these `Access Groups` must have at least `Read`

permission to the tenants. If neither of these Access Groups maps to a Security Role in iWD Manager, then the user will have full access to the iWD Manager GUI.

- If you do not want the user to have access to the `SYSTEM` tenant in iWD Manager, but only to an iWD managed tenant, then they must belong to an Access Group under the Configuration Server child tenant that maps to a Security Role in iWD Manager. In this case, they would not need to belong to any Access Group under the Environment tenant.
- If the user is created under the Environment tenant—for example, such as the default user that is in the Configuration Server database when it is initially deployed—then in order for this user to have full control of the Global Task List (not just read-only access), two conditions must be met:
 - The user must have a valid `Place` configured, where the `Place` is created under a child Configuration Server tenant to which the Interaction Server is associated.
 - The user must have an `Employee ID` that also belongs to a user who is created under a child Configuration Server tenant to which the Interaction Server is associated.

Examples

The following examples describe how this configuration would work in practice.

- Example 1a**
- You have a child tenant in Configuration Server called `ACME`. In iWD Manager that tenant maps to an iWD Managed tenant called `IWD_ACME`.
 - Under the `ACME` tenant, you have several `Places` configured: one for each user you want to log into iWD Manager and view the Global Task List. One of these `Places` is called `iwd_place3`.
 - Under the `ACME` tenant, you have a user with `User Name = system` and `Employee ID = system`. The user is a non-agent. On the `Annex` tab of the user object, you have configured the necessary section called `iWD` with the option name set to value `iWDManagerPlace` and the value set to `iwd_place3`.
 - This user belongs to an Access Group under the `ACME` tenant, called `iWDAdministrators`. He does not belong to any other Access Groups under the `ACME` tenant or the Environment tenant.
 - The `iWDAdministrators` Access Group has `Read & Execute` permissions to the `iWD Manager Application` object in Configuration Server.
 - The `iWDAdministrators` Access Group has `Read` permissions to the `ACME` tenant.
 - The `iWDAdministrators` Access Group maps to a security role in iWD Manager that gives the system user the necessary permissions for the iWD Manager GUI.

- Result: The system user can log into iWD Manager and access the IWD_ACME managed tenant. He cannot access the SYSTEM tenant.
- Example 1b**
- The system user described in Example 1a also belongs to the Administrators Access Group under the Environment tenant in Configuration Server. This Access Group has at least Read access to the Environment tenant.
 - In this case, the user can log into iWD Manager and access both the SYSTEM tenant and the IWD_ACME managed tenant.
 - You remove the user from the iWDAdministrators Access Group and instead make him a member of the Administrators Access Group under the ACME tenant in Configuration Server. The Administrators Access Group under the ACME tenant has Read access to the ACME tenant and Read & Execute permissions to the iWD Manager application object in Configuration Server.
 - Now, because the user belongs to two Access Groups that give them Read access to both the Environment tenant and ACME tenant, they can log into iWD Manager and access both the SYSTEM tenant and the IWD_ACME managed tenant. They need not belong to an Access Group that maps to an existing security role in iWD Manager.
 - Because they have Read permission to the Environment tenant in Configuration Server, this overrides any security policy that is configured for them in iWD Manager.
- Example 2a**
- You have a user configured under the Environment tenant. In this case, it is the default user who is set up when you first deployed Configuration Server. Their User Name is set to default and his Employee ID is set to default.
 - You have an unused Place configured under the ACME tenant called iwd_place4.
 - On the Annex tab of the user object, you have configured the necessary section called iWD with the option name = iWDManagerPlace and the value = iwd_place4
 - The user belongs to an Access Group that has at least Read permissions to the Environment tenant. He does not belong to any Access Groups under the ACME tenant.
 - Result: The default user can log into iWD Manager and have full access to the application, except the Global Task List. He can access the Global Task List in a read-only mode.
- Example 2b**
- You want the default user to also have full access to iWD Manager, including the ability to not only view tasks in the Global Task List under the IWD_ACME managed tenant, but also to manage the tasks through the Global Task List (Cancel, Hold/Resume, Modify)

- You change his Employee ID to an Employee ID of a user who is configured under the ACME tenant, such as the Employee ID of the user system configured in Example 1.

Note: This Employee ID cannot be shared by two users who will need to access the Global Task List simultaneously. This is because iWD Manager will be accessing Interaction Server on behalf of this user's Employee ID.

iWD Manager Login

Before accessing iWD Manager functionality, you have to log into the application. The credentials that are entered are authenticated against Genesys Configuration Server. Therefore, anyone who will need access to iWD Manager will need to be configured as a Person in Genesys Configuration Manager (also known as a User in Genesys Administrator).

Note: In order for iWD users to have access to the Genesys Configuration Server objects (for example, to access a list of skills from within a rule action), they must have permission to access the specific objects individually, or at the folder level (for example, access to the entire Skills folder in Configuration Server). For users to be allowed to access those objects, they must be members of some Access Group that has at least Read permission to those objects. This could be, but but this does not necessarily need to be, the Users Access Group. Alternatively, users may be added to the Security tab of these objects directly, rather than inheriting permission by virtue of the Access Group(s) of which they are a member.

The login screen (see [Figure 19](#)) prompts for the following information:

- Username—the username for the Person or User as configured in Genesys Configuration Server.
- Password—the password for the Person or User as configured in Genesys Configuration Server.
- CME Application—the name of the iWD Manager Application object, as configured in Genesys Configuration Server.
- Config Server Host—the name of the host on which Genesys Configuration Server is running.
- Config Server Port—the port Genesys Configuration Server is using.

The image shows the login screen for the Intelligent Workload Distribution (iWD) Manager. At the top, the text "intelligent Workload Distribution" is displayed in a light gray font. Below this is a blue-bordered box with a blue header labeled "Login". Inside the box, there are five input fields: "Username" (empty), "Password" (empty), "CME Application" (containing "iWDManager"), "Config Server Host" (containing "mcr800vladp"), and "Config Server Port" (containing "2020"). Below these fields is a blue button with a user icon and the text "Login". At the bottom of the screen, the Genesys logo is displayed, consisting of a red circular icon and the word "GENESYS" in a bold, sans-serif font.

Figure 19: iWD Manager Login Screen

Logging into iWD Manager Programmatically

You can log into iWD Manager programmatically by providing the URL for iWD Manager login page along with a valid username and password. This can facilitate a single sign-on process.

The URL format to use is:

```
http://<appserverhost>:<appserverport>/<iwdmanagerapplication>/ui/Login
.jsf?username=<username>&password=<password>&application=<CMEApplication>
&passwordEncoded
```

Where:

- <appserverhost> and <appserverport> are the host and port for the application server where iWD Manager is deployed.
- <iwdmanagerapplication> is the iWD Manager application running on the application server (for example, iwd_manager).
- <username> and <password> are a valid username and password combination for the user logging into iWD Manager.

- `<CMEApplication>` is the name of the iWD Manager application as configured in the Genesys Configuration database. You can find the name in Genesys Administrator or Configuration Manager. iWD Manager will be shown with an application type of Third Party Application.

Note: If you want include the `&passwordEncoded` at the end of the URL, then the value for `<password>` should be encoded using BASE64 algorithm. Otherwise, you can pass a plain-text password in the `<password>` parameter and omit `&passwordEncoded`.

An example URL is:

```
http://myTomcatHost:8080/iwd_manager/ui/login.jsf?username=jsmith&password=myPassword
```

iWD Manager User Interface

The iWD Manager user interface, shown in [Figure 20](#), is composed of three main application areas:

- Header bar (see [page 104](#)). Located on top, it shows general information and actions, such as logged-in user name and logout action.
- Navigational panel (see [page 104](#)). Located on the left-hand side, it provides navigation among the various configuration/management objects accessible in iWD Manager.
- Details view (see [page 107](#)). The largest area that is on the screen, it is where all of the details about an item that is selected in the tree can be viewed and modified.

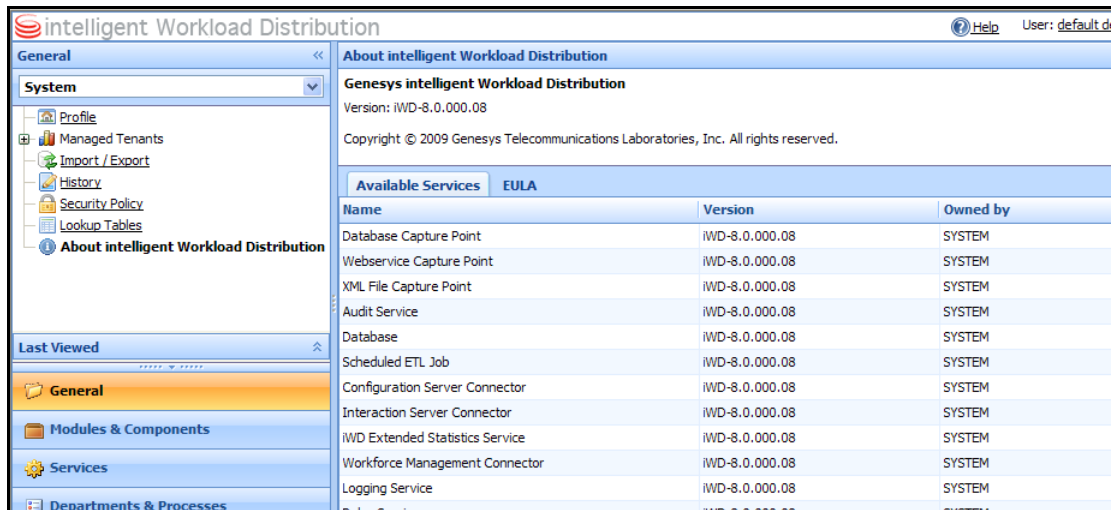


Figure 20: iWD Manager User Interface

Header Bar

The iWD Manager Header bar (see [Figure 21](#)) provides access to the following information and actions (left to right):

- Application logo.
- Undeployed-changes notification: This is shown only when there are changes in the current tenant's configuration that have not been deployed to runtime; such changes are not active. The notification also includes direct links to a deployment screen for each affected solution.
- Help: Opens the iWD Manager Help.
- User: Displays the first and last name of the currently logged-in user.
- Logout: Logs the user out of the iWD Manager application.

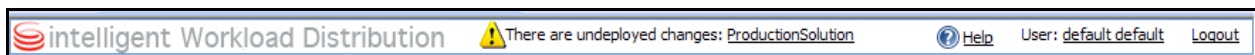


Figure 21: iWD Manager Header Bar

Navigation Panel

The iWD Manager Navigation panel (see [Figure 22](#)) consists of the following UI objects:

- Tenant selection: On top of the navigation area; allows switching among iWD tenants. See “Tenant Selection” on [page 105](#).
- Configuration sections: At bottom of the navigation area; allows switching among top-level configuration sections. See “Configuration Sections” on [page 106](#).
- Navigation tree: In the middle, provides access to configuration objects for the selected tenant. The objects shown depend on which navigation section is selected. See “Navigation Tree” on [page 107](#).
- Last Viewed: Between the navigation tree and its sections, provides quick access to the last-viewed panes. See “Last Viewed” on [page 107](#).

The Navigation area can be temporarily hidden by clicking the hide (<<) icon on the top-right corner; a hidden area can be made visible again by clicking the show (>>) icon.

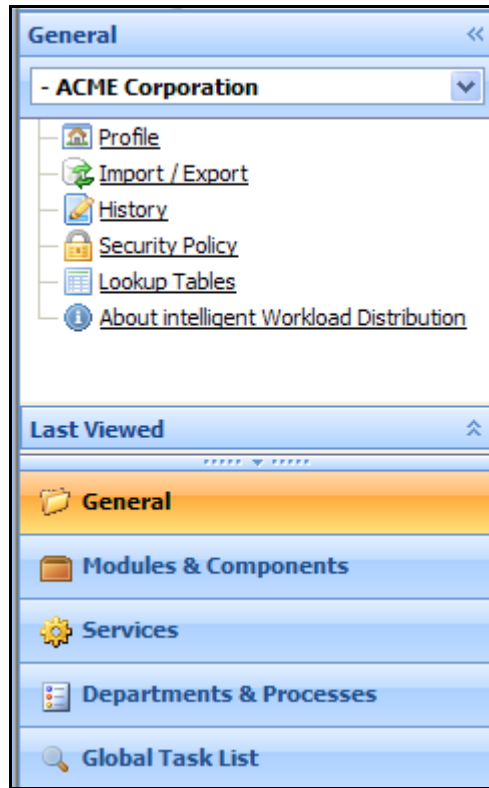


Figure 22: Navigation Panel

Tenant Selection

iWD configuration supports multi-tenancy, in which each iWD Manager user can have access to one or more tenants, depending on their security role and user access permissions. If more than one tenant is accessible to the logged-in user, tenant selection allows switching among them. See [Figure 23](#).

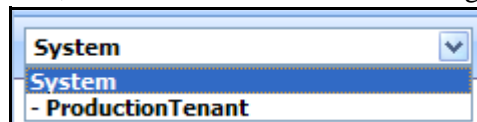


Figure 23: Tenant Selection

All configurations shown in different configuration sections are specific to the selected tenant.

Note: Each tenant in iWD configuration must be directly associated with a tenant in Genesys Configuration Server.

Configuration Sections

iWD Manager provides the following configuration sections (see [Figure 24](#)):

- **General:** General tenant configuration and actions, such as security policy and configuration import/export. For a detailed description, see “iWD Manager Overview” on [page 97](#).
- **Modules & Components:** Modules and components that are owned by or accessible to the tenant. For a detailed description, see “Modules and Components” on [page 125](#).
- **Services:** Solutions and services that are configured for a tenant. For a detailed description, see “Solutions and Services” on [page 136](#).
- **Departments & Processes:** iWD business configuration, such as departments, processes, and rules. For a detailed description, see “Departments and Processes” on [page 182](#).
- **Global Task List:** iWD management views. For a detailed description, see Chapter 5, “iWD Manager and Task Management,” on [page 207](#).



Figure 24: Configuration Sections

Depending on the application permissions of a logged-in user, only some of these sections might be visible. By default, upon logging in to iWD Manager, the General section is selected. To switch to another section, click on it in the navigation panel. The selected section is also displayed on the top of the navigation panel (just above the tenant selection).

Navigation can also be switched to an alternative, compact mode by clicking the collapse icon (on top). In compact mode, each navigation section will be represented by an icon. See [Figure 25](#).

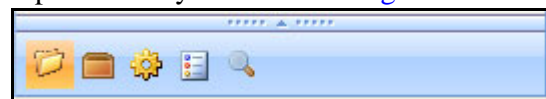


Figure 25: Compact Configuration Sections

To restore normal navigation mode again, click the expand icon (on top).

Navigation Tree

The navigation tree, shown in [Figure 26](#), displays iWD configuration and management objects for the selected tenant and navigation section in a tree structure. Each selectable object is underlined and can be selected by clicking it. When it is selected, the object is marked in bold, and the corresponding details are displayed in the iWD Manager Details area.

For those objects that support creation of new instances, a “New...” action also is present in the tree (such as “New Department...” in the example below). When a “New...” action is selected, details for the new object instance can be entered and saved in the iWD Manager Details area.

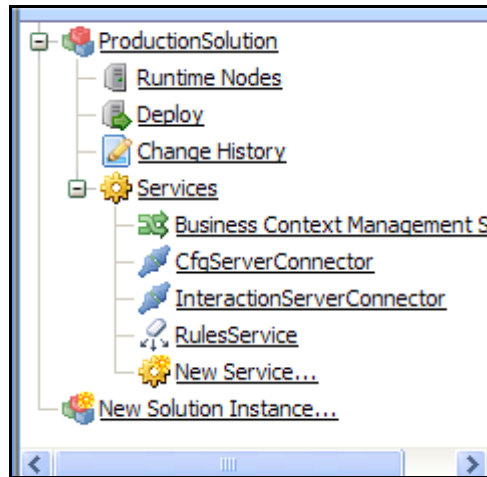


Figure 26: Navigation Tree

Last Viewed

By default, the Last Viewed list is hidden. It can be displayed and hidden again, similar to the whole Navigation section. See [Figure 27](#).

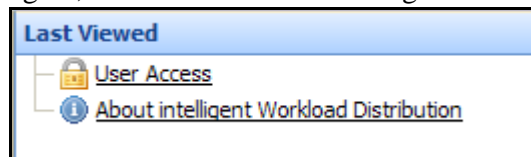


Figure 27: Last Viewed

When it is visible, Last Viewed lists the five last-viewed panes. The details for each item in the list can be opened directly from the Last Viewed list by clicking the corresponding item.

Details View

Details view shows and (for some objects) allows modification of information that is related to the object that is selected in the Navigation Panel (see

page 104). Figure 28 illustrates an example of the Details view for a department.

ProductionSolution > Financial Department

General | **Rules**

ID T2_C1 **Department Name** Financial Department **Start Date** 01/01/2001 **End Date** 12/07/2022

Description
Financial department. Statistics and lists need to be sent to the contact person every Monday morning.

Contact Name John Smith **Contact Email** smith@acme.com **Contact Phone** +1 415 1234567890

Metrics (historical reporting)

Template	Metric	Distribution point	Value	Description

Custom Attributes (historical reporting)

Name	Type	Value	Description

Save **Save & Close** **Cancel** **Delete** **View tasks**

Figure 28: Details View

Object Actions

Most of the iWD Manager objects can be modified, in which case the Save, Save & Close, Cancel, and Delete actions are available at the bottom of the Details views (see Figure 29):

- **Save:** Saves the current object, and leaves it selected.
- **Save & Close:** Saves and closes the current object.
- **Cancel:** Discards any modifications and closes the current object.
- **Delete:** Deletes the current object.
- **Copy:** iWD Manager has built-in copy and paste functionality for the following configuration objects: Services, Departments, Processes, Rules, and Business Calendars. Services, Rules, and Business Calendars are pasted as new objects that must be saved. Departments and Processes perform cascaded copy; child rules also are copied. Clicking Copy on the

edit page of an object puts it in the clipboard (only one object at a time is supported). When the clipboard contains an object, depending on the object type, additional tree nodes or buttons will be visible/enabled that allow pasting the clipboard object to a new instance.

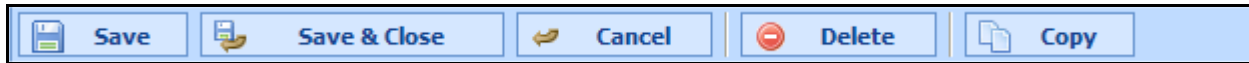


Figure 29: Details View Actions

Note: The ability to modify an object also depends on the user's permissions. If the user does not have permission to modify a particular object, the previously described object actions will not be accessible.

Unsaved Changes

iWD Manager keeps track of any changes that have been made to the current selected object by a user.

If an object is being closed, but unsaved changes are present, iWD Manager will display the Unsaved Changes dialog box, shown in [Figure 30](#). The dialog box has the following actions:

- Continue: Discards the modifications and closes the object
- Return: Returns to the Details view of the object and does not discard the modifications.

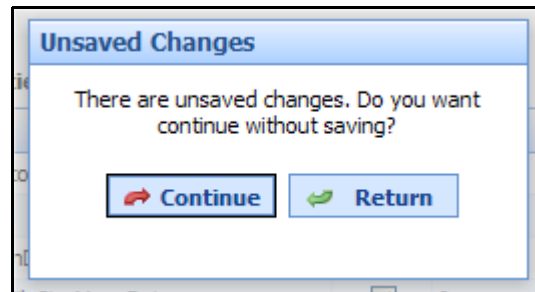


Figure 30: Unsaved Changes

Context History

In addition to warning about unsaved changes, iWD Manager also records all saved changes into the object's Context history. Context history can be viewed by clicking the Context History icon in the top-right corner of the object's Details view. [Figure 31](#) shows an example Context History for a Linear Rule.

The Context History appears in a new window and displays detailed information about all of the changes that have been made to the current object:

- Date/Time: When the modification was made.
- User: Who made the modification.

- Object Code: Describes the object type (such as RL, which means “Linear Rule”).
- Event Code: Describes the action type (such as UPD, which means “Update”).
- Event: Formatted description of the change.

Date/Time	User	Object Code	Event Code	Event
Aug 12, 2010 4:33:29	default	DPT	CRT	Department created: Financial Department

Details

Figure 31: Context History

Additionally, for UPD (update) events, the Context History provides details on each attribute of an object that has been changed. These details can be viewed by clicking a particular UPD event in the upper table. They are displayed in the Event pane.

User Profile

You can customize date/time conversions and iWD Manager appearance in the User Profile dialog box, shown in [Figure 32](#). To access the User Profile dialog box, click the user name in the Header bar (see “Header Bar” on [page 104](#)). If Timezone is not specified, all timestamps will be shown by using the UTC time

zone. The User Profile also allows you to specify your Locale, which determines the format that is used for dates, times, numbers, and currency.

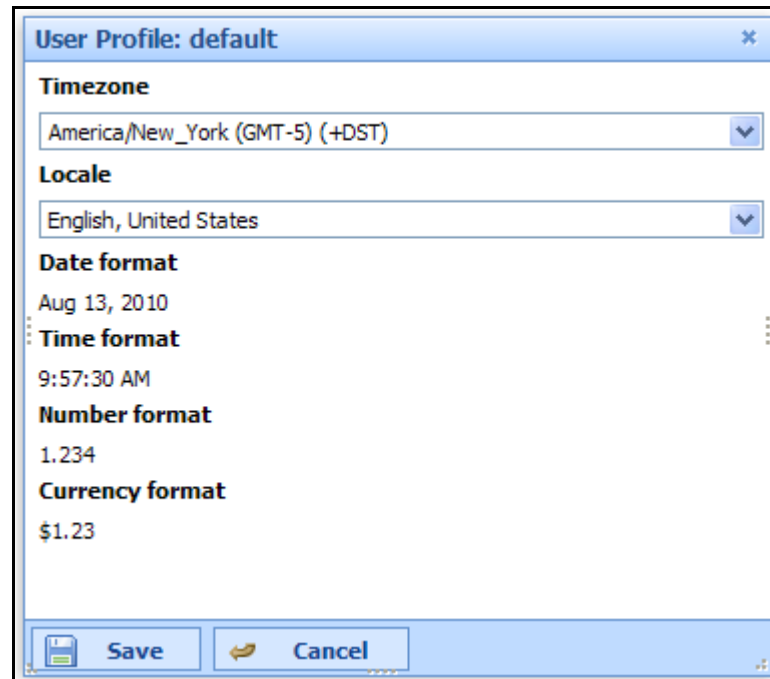


Figure 32: User Profile Dialog Box

General Configuration

The General configuration section (see [Figure 33](#)) provides the ability to manage general tenant-level configuration:

- Use Profile view (see [page 112](#)) to view and modify general tenant attributes.
- Use Import and Export view (see [page 114](#)) to import and export tenant's configuration to and from XML files.
- Use Security Policies view (see [page 116](#)) to manage tenant roles and permissions.
- Use Managed Tenant Details view (see [page 118](#)) to create new managed tenants, and view and modify attributes of managed tenants.
- Use Lookup Tables view (see [page 120](#)) to specify tenant-specific constants as key/label pairs.
- Use History view (see [page 123](#)) to keep track of changes in tenant configuration.

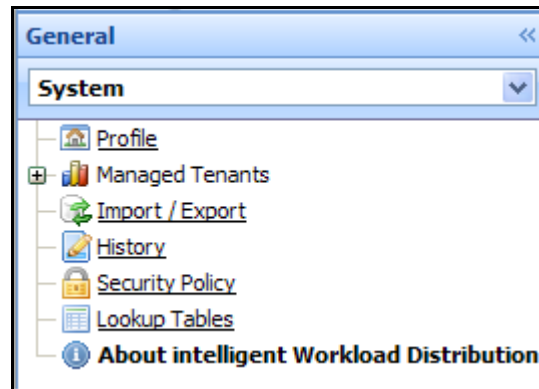


Figure 33: General Configuration Section

Profile

The Profile view (see [Figure 34](#)) displays and allows modification of general tenant details, such as name, description, and default time zone.

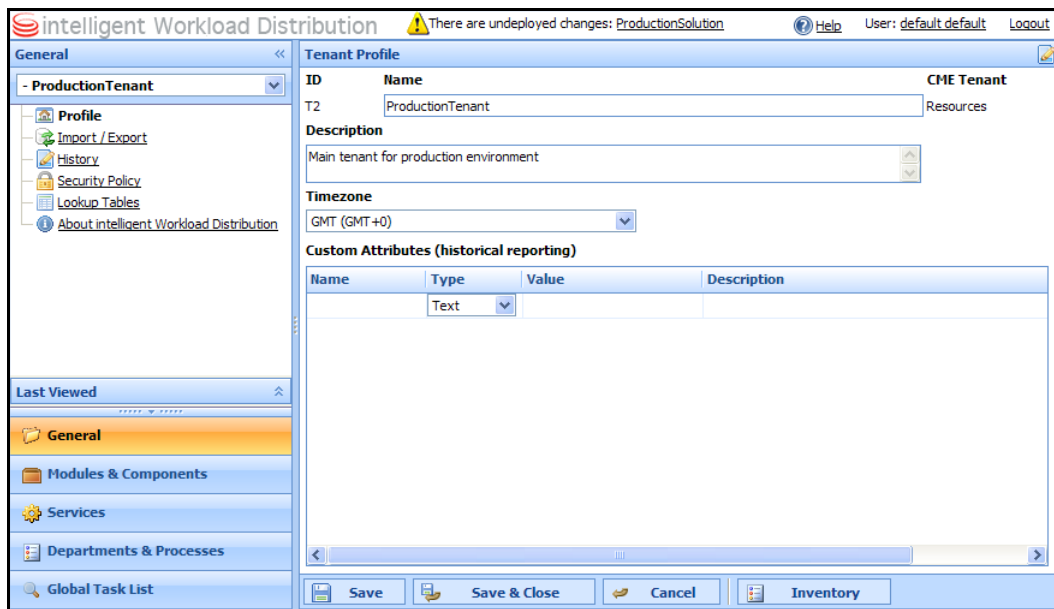


Figure 34: Profile View

[Table 1](#) lists the properties and actions available in the Profile view.

Table 1: Profile View Attributes and Actions

Property/Action	Description
ID	Tenant ID. This field is read-only after the first save.
Name	Tenant name. This field is only editable by parent tenant's administrators.

Table 1: Profile View Attributes and Actions (Continued)

Property/Action	Description
CME Tenant	<p>The name of the corresponding tenant in Genesys Configuration Server. The iWD System Tenant is associated with the Resources tenant in a single-tenant Configuration Server, and with the Environment tenant in a multi-tenant Configuration Server.</p> <p>When you create your iWD Tenant, select the corresponding Configuration Server Tenant from the drop-down list. This ensures that the user in iWD Manager has access to the Configuration Server data objects for which the user has permission.</p>
Description	Tenant description.
Time Zone	Tenant default time zone. The default time zone is used when no time zone is specified for a capture point.
Custom Tenant Attributes	Custom tenant attributes that provide additional information about the tenant for reporting purposes. Refer to the <i>iWD 8.0 Data Mart Reference Guide</i> for more information.
Save/Save & Close/Cancel	Standard iWD Manager functions as described in “Object Actions” on page 108 .
Inventory	Opens the tenant business process inventory report that contains detailed information about tenant business configuration (such as departments, processes, and rules). See Figure 35 for an example).

intelligent Workload Distribution Department & Business Process Inventory Report	
System: System	Created: Aug 13, 2010 7:41 AM
System ID: SYSTEM	
Tenant: ProductionTenant	
Tenant ID: T2	
Tenant Name: ProductionTenant	
CME Tenant Name: Resources	
Tenant Active: Yes	
Date Created: Aug 12, 2010 2:13 PM by default	
Date Last Updated: Aug 12, 2010 4:15 PM by default	
Solution: ProductionSolution	
Solution ID: SLT2	
Solution Name: ProductionSolution	
Solution Description: Solution for the production environment.	
Department: Financial Department	
Department ID: T2_C1	
Department Name: Financial Department	
Department Description: Financial department. Statistics and lists need to be sent to the contact person every Monday morning.	
Department Start Date: 31/12/2000	
Department End Date: 11/07/2022	
Contact Name: John Smith	
Contact Phone: +1 415 1234567890	
Contact Email: smith@acme.com	

Figure 35: Inventory Report

Import and Export

The Import/Export function enables you to import and export the iWD configuration to and from an XML file. This function allows you to:

- Transfer configurations between environments and tenants.
- Load the configuration from the solution template, an exported configuration of a solution that can be used as the basis for configuring a new solution

Figure 36 shows the Import/Export function.

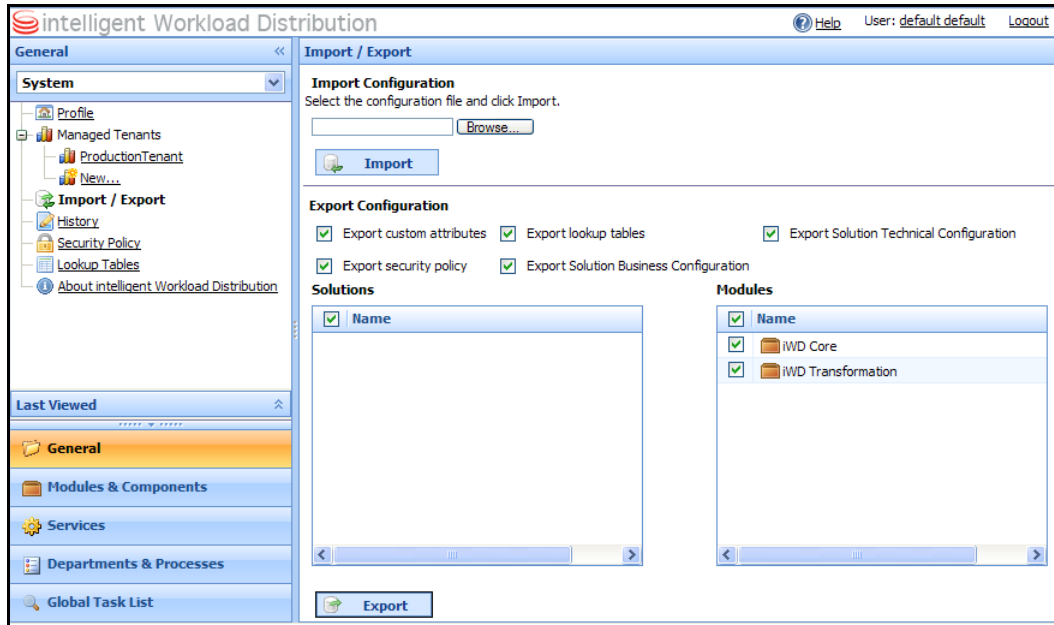


Figure 36: Import/Export

Table 2 lists the properties and actions available in the Import/Export view.

Table 2: Import and Export Properties and Actions

Property/Action	Description
Configuration File/Browse	Allows selection of a configuration file to import.
Import	Imports the configuration file that is selected in the field above.
Export custom attributes	Whether to include the tenant's custom attributes (see page 112) in the exported configuration file.
Export security policy	Whether to include the tenant's security policies (see page 116) in the exported configuration file.
Export lookup tables	Whether to include the tenant's lookup tables (see page 120) in the exported configuration file.
Export Solution Business Configuration	Whether to include the tenant's Solution Business Configuration (such as departments, processes, rules, and business calendars) in the exported configuration file.
Export Solution Technical Configuration	Whether to include the tenant's Solution Technical Configuration (such as runtime nodes and services) in the exported configuration file.

Table 2: Import and Export Properties and Actions (Continued)

Property/Action	Description
Solutions	Which of the tenant's solutions (see page 137) to include in the exported configuration file.
Modules	Which of the tenant's modules (see page 126) to include in the exported configuration file.
Export	Exports the configuration objects that are selected above to an XML configuration file.

Security Policies

A Security Policy (see [Figure 37](#)) allows you to create custom security roles for each tenant and map them to Genesys Configuration Server Access Groups. During authentication, a user is granted all of the permissions that are combined from roles that are mapped to the Genesys Configuration Server Access Groups of which the user is a member. A role can contain any combination of permissions from three groups: administrative permissions (technical configuration), configuration permissions (business user configuration), and task management permissions (the Global Task List view in iWD Manager). For them to function properly, most permissions must be added in groups; for example, modify or delete permissions are useless without the view permission.

Users (Person objects) are assigned to these groups in Genesys Configuration Server. Any person assigned to a group that you choose to map to under Group Mapping will have all the selected permissions within iWD Manager. You

might have to create multiple new Access Groups in Genesys Configuration Server so the appropriate mapping can be done in the Security Policy dialog.

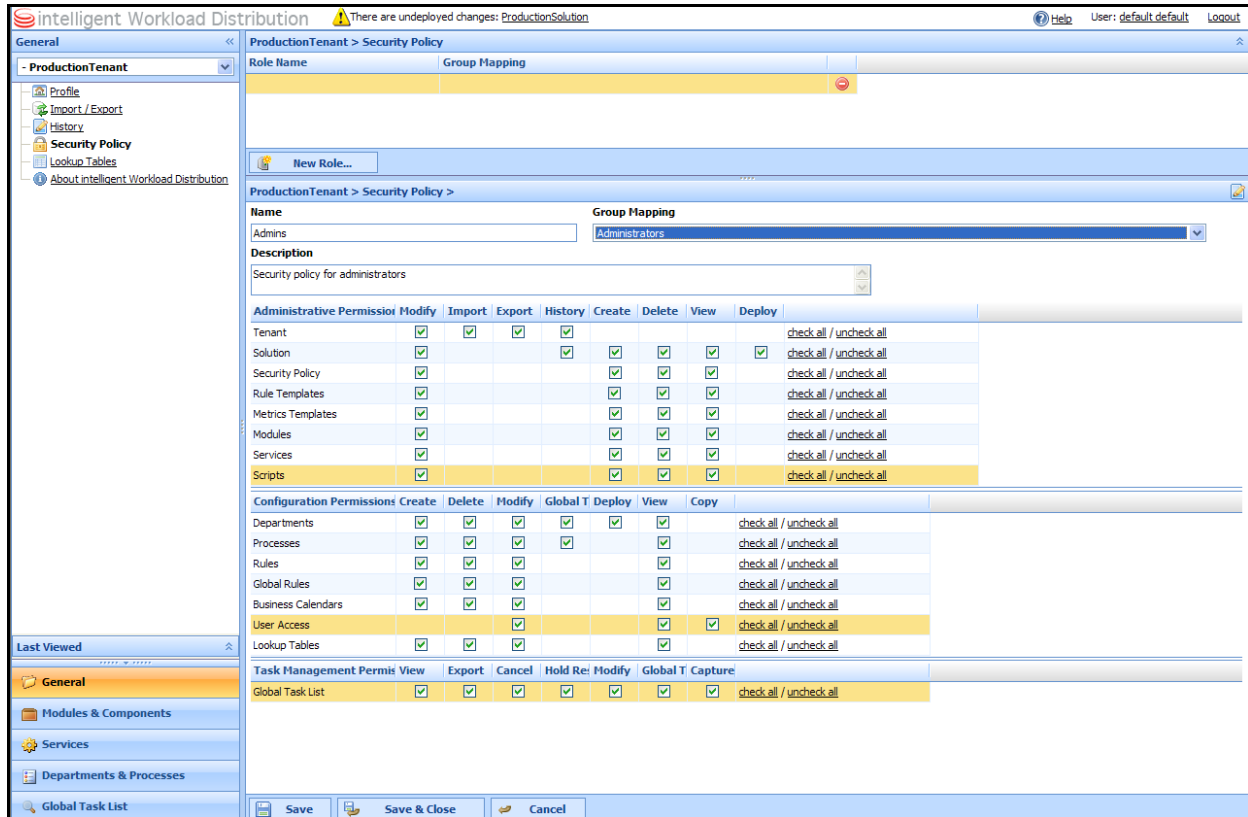


Figure 37: Security Policies

Key features:

- Multiple roles can be mapped to the same Genesys Configuration Server Access Group.
- Any Person or User in Genesys Configuration Server who is a member of the Administrators or Super Administrators access group in the Environment Tenant has non-restricted access to all tenants.
- Automatic Access Group lookup from Genesys Configuration Server.

Table 3 lists the properties available for Security Policies.

Table 3: Security Policy Properties

Property	Description
Name	Role name. This is a mandatory field and must be unique within the tenant.

Table 3: Security Policy Properties (Continued)

Property	Description
Group Mapping	Configuration Server Access Group mapping. This is a mandatory field. Select the Configuration Server Access Group from the drop-down list.
Description	A plain text description of the role (up to 4,000 characters).

Note: In iWD 8.0, iWD maintains its own security roles. These roles are independent from Roles that are defined in Genesys Administrator. That is, for a single user defined in the Genesys System, they may be assigned one or more roles in Genesys Administrator, as well as inheriting one or more roles that have been defined in iWD Manager. The roles in Genesys Administrator are used to define which tasks a user can perform in specific applications, such as Genesys Administrator itself as well as Genesys Interaction Workspace. The roles in iWD are used to define what tasks a user can perform in iWD Manager.

Managed Tenant Details

The Managed Tenant Details view (shown in [Figure 38](#)) displays and allows you to modify general managed tenant attributes (see “Tenants” on [page 29](#)).

Managed Tenants > ACME

ID: T2 Name: ACME CME Tenant: mcr800

Description:

Assigned Modules:

- ☐ IWD Core
- ☐ IWD Database Capture
- ☐ IWD Datamart
- ☐ IWD Reporting
- ☐ IWD Rules
- ☐ IWD Webservice Capture
- ☐ IWD Transformation
- ☐ IWD XML File Capture

Unassigned Modules:

Save Save & Close Cancel Delete

Figure 38: Managed Tenant

[Table 4](#) lists the attributes and actions available in Managed Tenant view.

Table 4: Managed Tenant Properties and Actions

Property/Action	Description
ID	Tenant ID. This field becomes read-only after a tenant is created.
Name	Tenant name.
CME Tenant	<p>The name of the corresponding tenant in Genesys Configuration Server.</p> <p>When creating your iWD Tenant, select the corresponding Configuration Server Tenant from the drop-down list. This ensures that the user in iWD Manager only has access to the Configuration Server data objects for which he or she has permission.</p> <p>In a single-tenant environment the configured managed tenant must map to the <code>Resources</code> tenant in Genesys Configuration Server.</p> <p>An iWD Managed Tenant may have a one-to-one relationship with a tenant in Genesys Configuration Server.</p>
Description	Tenant description.
Assigned Modules	Modules (or functionality) that are assigned to the tenant. Such modules are available for use in tenant solutions.
Unassigned Modules	Modules that are not assigned to the tenant. Such modules are not available for use in tenant solutions.
Save/Save & Close/Cancel/Delete	Standard iWD Manager functions as described in “Object Actions” on page 108 .

Lookup Tables

Users can specify lookup tables that can be used in rules, custom attributes, and metrics. Lookup tables are simple key/label pairs and are displayed as dropdown controls. See [Figure 39](#).

Lookup Tables	
Lookup Table	
channels	-
Lookup1	-

New Lookup Table

Lookup Table > channels

Name
channels

Values

Key	Label	+	-
e-mail	E-mail	+	-
fax	Fax	+	-
web	Web	+	-
letter	Letter	+	-
virtual task	Virtual Task	+	-

Figure 39: Lookup Tables

Distribution Points and Lookup Tables

The concept of Distribution Points has changed in iWD 8.0. They are no longer required, but to keep reporting functionality intact they remain as configuration objects in iWD (as opposed to the Services they were in earlier releases).

Distribution points should be configured as Lookup Tables (see [page 120](#)) at the Tenant level. The following procedure describes the steps used to configure Distribution Points as Lookup Tables.

Procedure: Configuring Distribution Points as Lookup Tables

Purpose: To configure Distribution Points as Lookup Tables and assign them to Rules.

Start of procedure

1. In iWD Manager, configure a new Lookup Table for your tenant as shown in Figure 40. The name of the Lookup Table **must** be `distributionPoints`.

Note: The `distributionPoints` lookup table must be configured under the iWD managed tenant in which the rule action to assign the distribution point to a task is defined.

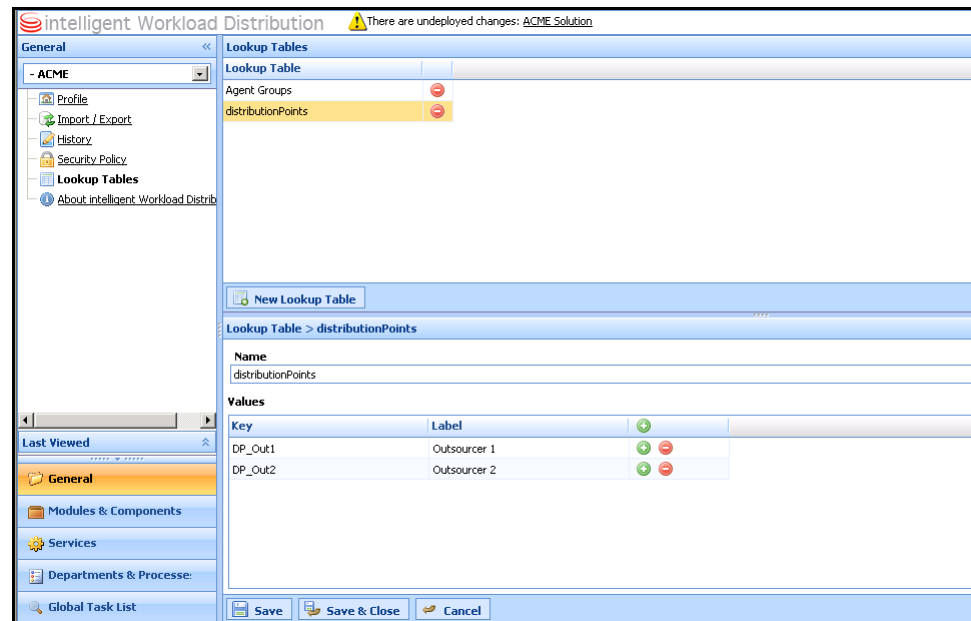


Figure 40: Create a Lookup Table

2. In a new or existing Rule Template, add the newly created Lookup Table (Distribution Point) as a parameter, as shown in [Figure 41](#).

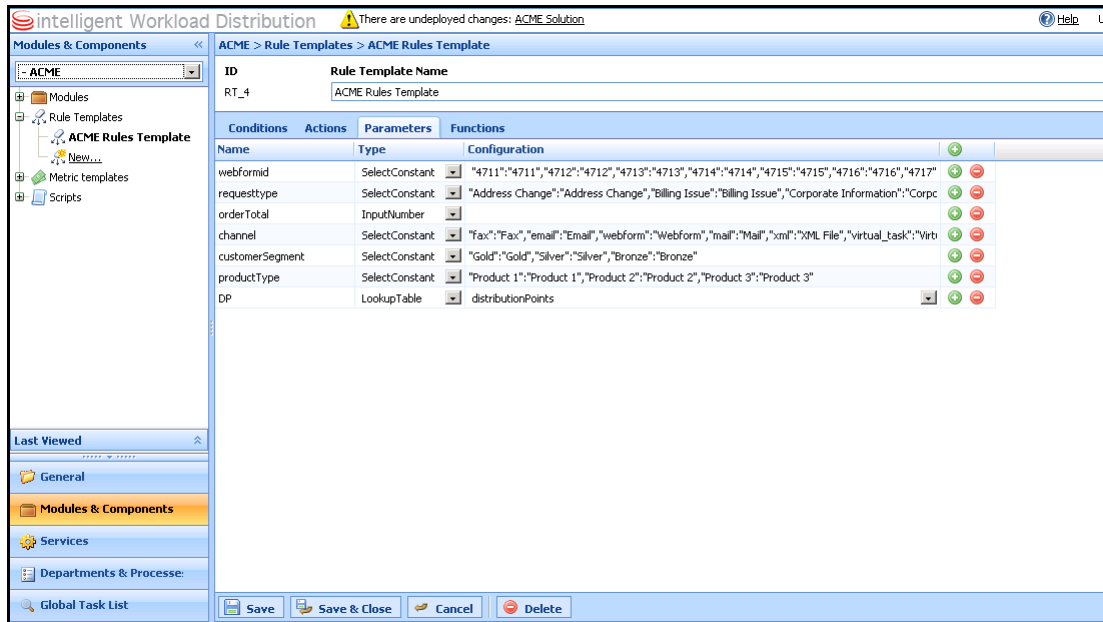


Figure 41: Add as a Parameter to the Rule Template

3. Add an Action. You can copy and paste the Language Expression and Rule Language Mapping from the rule action Assign distribution point that is part of the Standard Rules Template. If you have created a new rule template, follow the procedure [Procedure: Adding a Rule Template to your Solution](#), on [page 128](#). See [Figure 42](#).

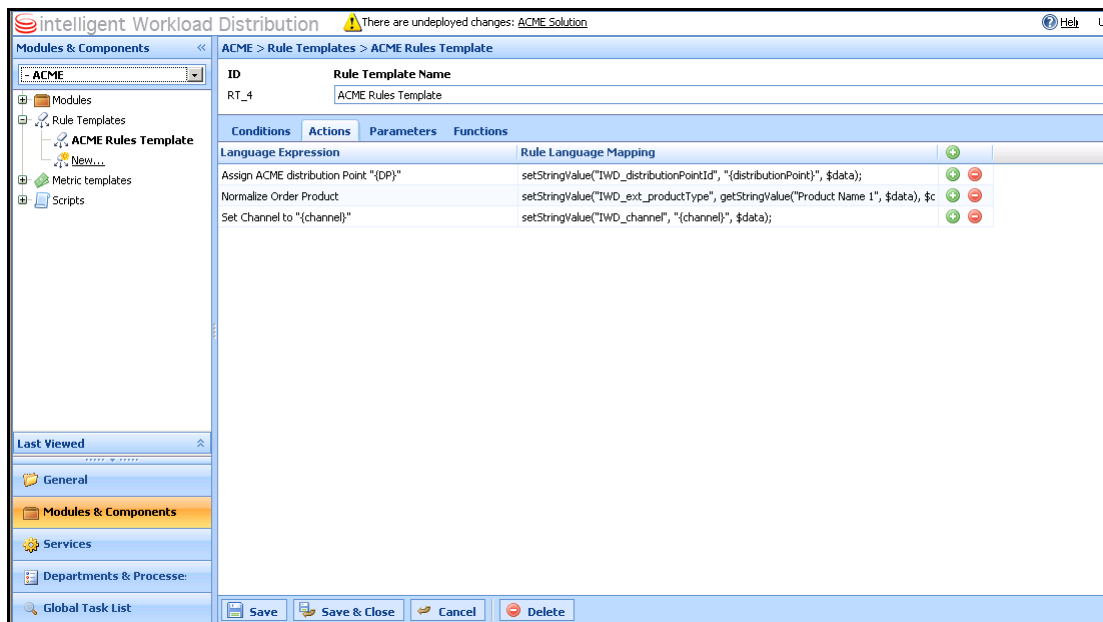


Figure 42: Add an Action

- Once the rule template has been updated with the new action, the Distribution Point can be assigned in Classification Rules, as shown in [Figure 43](#).

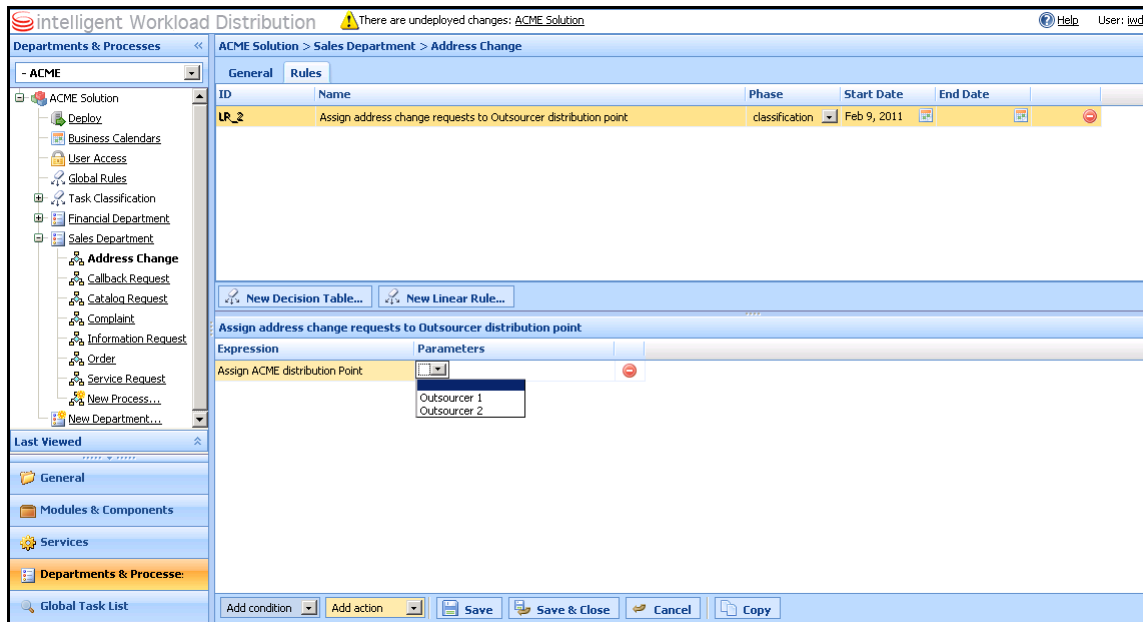


Figure 43: Classification Rules

End of procedure

Next Steps

It is possible to configure separate sets of Distribution Points for each Solution:

- On the Tenant level, several Lookup Tables should be created, each with its own set of Distribution Points.
- Several Rule Templates should be created. Each has to use its own Lookup Table.
- On the Tenant level, several Modules should be created and corresponding Rule Templates should be assigned.
- On the Solution level, corresponding Modules should be assigned to the particular Solution.

Each Solution will “see” its own set of Distribution Points during Rules creation.

History

The History page (see [Figure 44](#)) provides a detailed log of activities that have been performed in iWD Manager by users. Each activity is represented by an

audit event that represents what was changed, by whom, and when the change was made.

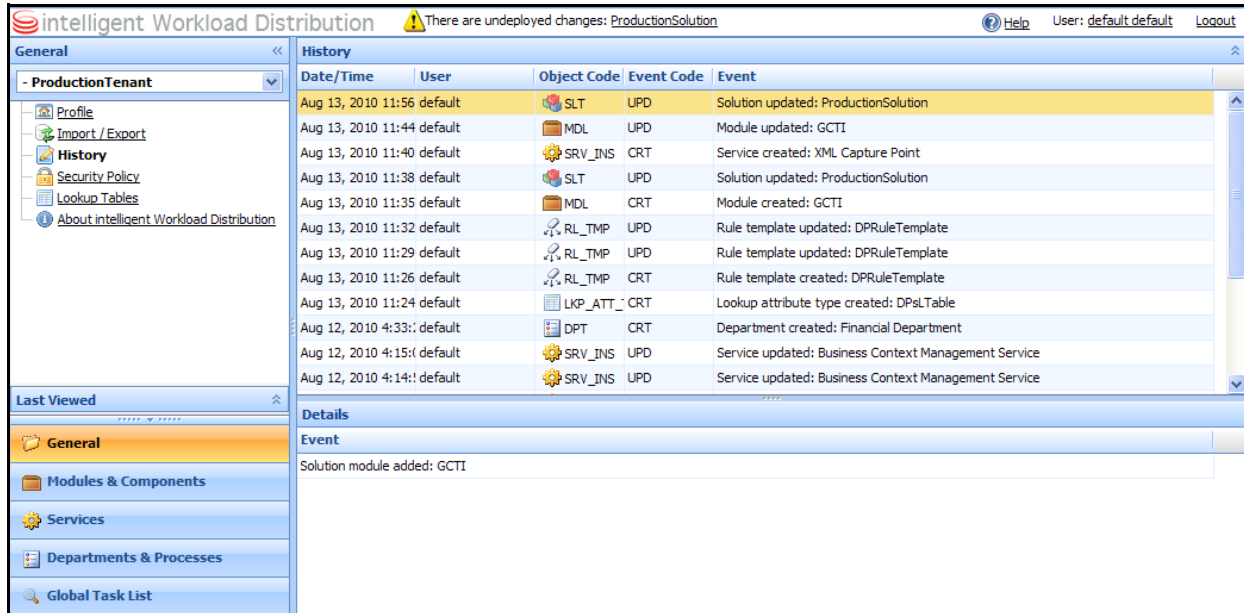


Figure 44: History

Table 5 lists the properties and actions are available in History view.

Table 5: History View Properties and Actions

Property/Action	Description
Date/Time	Date/time when the activity was performed.
User	Who performed the activity.
Object Code	Represents the type of object on which the activity was performed: <ul style="list-style-type: none"> TNT: Tenant SLT: Solution SRV_INS: Service Instance RNT_ND: Runtime Node RL_TMP: Rule Template MDL: Module PRC: Process RL: Rule DPT: Department ROLE: Security Policy Role BC: Business Calendar

Table 5: History View Properties and Actions (Continued)

Property/Action	Description
Event Code	Represents the type of activity: <ul style="list-style-type: none"> • CRT: Create • UPD: Update • DLT: Delete • IMP: Import • EXP: Export
Event	Formatted description of the activity.
Details	Displays attribute-level changes to the object for the activity that is selected in the main History table. Only update activities (Event Code is UPD) contain these details.

Modules and Components

The Modules and Components configuration section is used to define objects for iWD functionality:

- A component is an object that provides a specific iWD function. The following component types are used in iWD:
 - Service templates: This component provides a template for an iWD service that implements specific functionality. Service templates are preconfigured and are not changeable in iWD Manager. Service instances (see [page 139](#)) that are based on service templates, however, are configurable.
 - Rule templates (see [page 127](#))
 - Metrics templates (see [page 135](#))
- A module (see [“Modules”](#)) is a group of components.

Assigning Modules to Tenants and Solutions

Whenever you create a new Tenant or a new Solution under a Tenant, be sure to verify the Assigned Modules are correct for the Tenant or Solution.

After you create a new Tenant, go to the General section in iWD Manager. Select the System tenant from the tenant selection drop-down list. In the navigation tree, locate your new Tenant in the Managed Tenant tree. On the right hand side of the screen, be sure that the Assigned Modules list includes all Modules to which you want the Managed Tenant to have access.

After you create a new Solution under that Managed Tenant, select that Managed Tenant from the tenant selector, and then click on the name of your

Solution in the navigation tree. On the right hand side of the screen, make sure you assign the Modules you want to use as part of that Solution.

Modules

Modules bundle a set of iWD components into a named functional area. Modules are used to group “features”, such as capture from WebSphere, reporting, and so on. Each feature maybe be implemented by one or more components where the component can be a service, a rule template, a metric template and a (transformation) script.

This is done in order to:

- simplify the exporting/importing of configuration that enables particular functions.
- simplify the enabling/disabling of a particular function to a tenant.

Figure 45 shows an example.

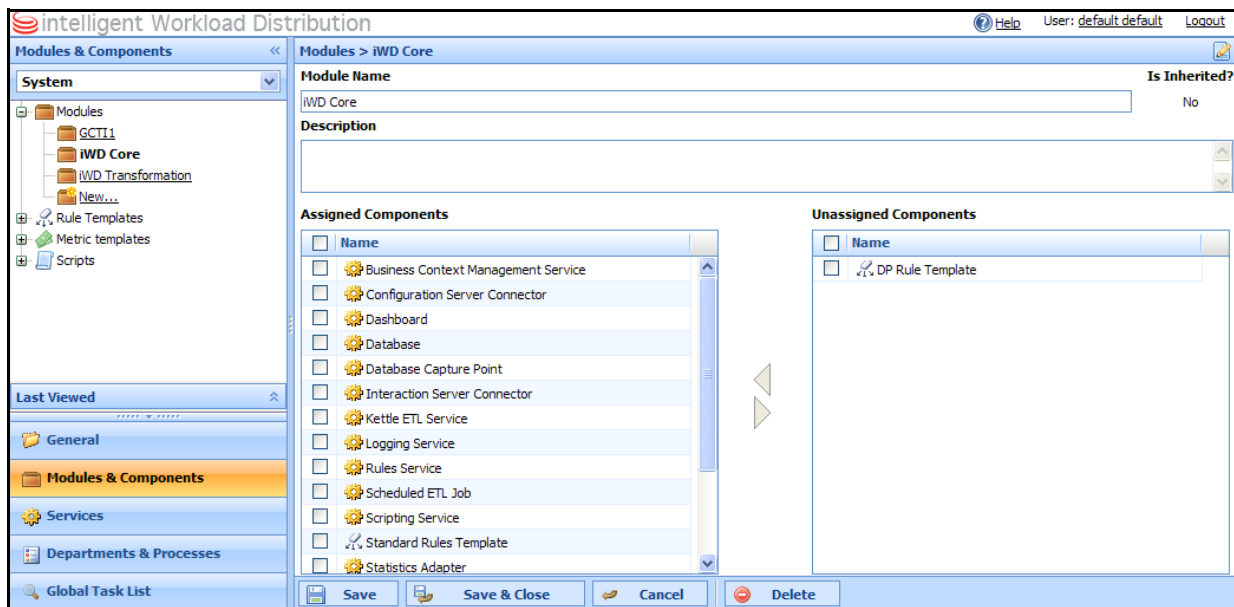


Figure 45: Module

[Table 6](#) lists the properties and actions available in the Modules view:

Table 6: Modules View Properties and Actions

Property/Action	Description
Name	Name of the module
Is Inherited	Indicates whether a module is inherited from the parent tenant. This means that the module has been assigned to this managed tenant by the parent tenant. If this is true, all fields are read-only. All standard modules in iWD belong to the System tenant, and are inherited by child tenants.
Description	Description of the module.
Assigned Components	Services, rule templates, metrics templates, and scripts that belong to the module.
Unassigned Components	Services and rule templates that do not belong to the module.
Save/Save & Close/Cancel/Delete	Standard iWD Manager functions as described in “Object Actions” on page 108 .

Rule Templates

In the iWD solution, Rule templates are used to set up different business rules for task classification, department, and processes, as well as rules at the global level. Business rules usually have the following:

- Conditions—“when” or “if” expressions
- Actions—“then” expressions (required)
- Parameters—used in the actions and conditions

Business rules within iWD are based on Rule templates that are provided out-of-the-box. The out-of-the-box Rule templates are maintained by IT personnel. IT personnel may create new Rule templates and modify out-of-the-box templates to meet business requirements. Business rule templates provide the foundation for business users to create business rules that govern task handling in iWD. Business users are empowered to make the necessary changes to business rules while IT personnel manage the rule template.

Note: Definition of Rule templates requires a basic knowledge of the Java programming language.

After you create a new Rule Template, you have to explicitly add it to your Solution. This outlined in the following procedure.

Procedure:

Adding a Rule Template to your Solution

Purpose: New Rule Templates need to be explicitly added to your Solution before you can use them. This procedure outlines how to add the Rule Template to your Solution.

Start of procedure

1. Create the new Rule Template (Modules & Components > Rule Templates > New) and save it (refer to [Table 7](#) for the configurable properties). Your new Rule Template will become a new Component, which needs to be added to a Module.
2. Add the new Rule Template Component to an existing Module, or create a new Module and add it there. Do this by going to Modules & Components > Modules. You should see your new Rule Template in the Unassigned Components list. Check it and click the arrow to move it to the Assigned Components list. Save the change.
3. If you used an existing Module in [Step 2](#), and that Module is already assigned to your Solution, you should be done. If you created a new Module in [Step 2](#), then you need to add this Module to your Solution. Go to Services, select your Tenant, and then click on the name of your Solution. There you will see a list of Assigned Modules and Unassigned Modules. Select your new Module (with your Rule Template in it) from the list of Unassigned Components. Click the arrow to move it to the Assigned Modules list. Save the change.
4. If you added the Rule Template to an existing Module, you do not need to redeploy your Solution in order to use the new template. If you created a new Module in [Step 2](#), you will be prompted to deploy the Solution first.

Note: These steps assume you have created the Rule Template under a specific iWD Tenant. You can follow the same steps to create a new template under the System Tenant, if desired.

End of procedure

[Table 7](#) lists the properties available in Rule Template view.

Table 7: Rule Template View Properties

Property	Description
Rule Template Name	The name of the rule template. This field is mandatory.
Conditions	The rule condition template; that is, a <i>when</i> or <i>if</i> expression (for example, <i>when due time is in x hours</i>). The Rule template can contain any number of (or no) conditions. See “Conditions and Actions” .
Actions	The rule action template; that is, a <i>then</i> expression (for example, <i>then assign priority</i>). The Rule template can contain any number of actions. See “Conditions and Actions” .
Parameters	Rule parameters that are used in conditions and actions. These describe how parameters are presented to the user (who is creating the rule), how input is validated, and so on. See “Parameters” on page 131 .
Functions	Allows definition of more complex functions, which then can be referenced in actions and conditions. See “Functions” on page 133 .
Save/Save & Close/Cancel	Standard iWD Manager functions as described in “Object Actions” on page 108 .

Conditions and Actions

Although conditions and actions have different purposes, their definition is the same. Table 8 lists the property specifications.

Table 8: Conditions and Actions Properties

Property	Description
Language Expression	Semantics of the rule language; meant for business people. They are simple enough to be understood by non-Java specialists. Language expressions can contain different parameters. Warning: Do not use the word <code>end</code> in rule language expressions. Using the word <code>end</code> causes rule parsing errors.
Rule Language Mapping	The same rule language expression, in Java code. This is the real code that is executed; it requires an understanding of Java and JBoss rules. JBoss is the underlying rule engine used in iWD.

Figure 46 shows an example of conditions for a Rule template.

Language Expression	Rule Language Mapping
Business value is "{businessValue_From}" to "{businessValue_To}"	<code>eval({businessValue_From} <= getIntValue("IWD_businessValue", \$data) && {businessValue_To} >= getIntValue("IWD_businessValue", \$data));</code>
Channel is "{taskChannels}"	<code>eval(getStringValue("IWD_channel", \$data).equals("{taskChannels}"));</code>
Department is "{department}"	<code>eval(getStringValue("IWD_departmentId", \$data).equals("{department}"));</code>
Due Time is in "{periodFrom}" to "{periodTo}" "{periodType}"	<code>eval(getDTValue("IWD_dueDateTime", \$data)!=null && getDTValue("IWD_dueDateTime", \$data).after(getPeriodDTFrom(getCurrency("IWD_currency"), {periodFrom}, {periodTo}, {periodType}));</code>
Integer "{attribute}" "{operator}" "{integerValue}"	<code>eval(compareInteger(getIntValue("{attribute}", \$data), "{operator}", {integerValue});</code>
Is first prioritization	<code>eval(!\$data.containsKey("IWD_reprioritizeDateTime"));</code>
Is reprioritization	<code>eval(\$data.containsKey("IWD_reprioritizeDateTime"));</code>
Media type is "{mediaType}"	<code>eval(getStringValue("Media type", \$data).equals("{mediaType}"));</code>
No process selected	<code>eval(isNull("IWD_processId", \$data));</code>
Priority is "{operator}" "{priority}"	<code>eval(compareInteger(getIntValue("Priority", \$data), "{operator}", {priority});</code>
Process is "{process}"	<code>eval(isProcess("{process}", \$data));</code>
String "{attribute}" equals "{stringValue}"	<code>eval(getStringValue("{attribute}", \$data).equals("{stringValue}"));</code>
Task is overdue	<code>eval(getDTValue("IWD_dueDateTime", \$data)!=null && getDTValue("IWD_dueDateTime", \$data).before(getCurrencyDT(\$data));</code>

Figure 46: Rules Template—Conditions Tab

Reviewing the third condition, the language expression contains the parameter `{department}`, which is mapped to the rule language. This expression is executed for each task to which this rule can apply, as follows: the parameter is replaced with the current parameter value, which is compared to the task department value, which in turn is retrieved by calling a Task object method and converted to an integer value. During the building of rules, only the

language expressions are visible to business users. This enables them to build rules that are easy to read and understand.

Note: Rule language mapping follows JBoss Rules syntax (which is Java with some minor extensions). Regarding available methods, there is a single variable available to rules called `$data` which represents task attributes as a key-value collection. It is an instance of the `KeyValueCollection` class which is part of Platform SDK (refer to the *Platform SDK 7.6 (or later) Developer's Guide* for more information).

Parameters

Rule-language expressions and the corresponding language mappings (used in conditions and actions) can contain different parameters. For example, the expression `Process is {process}` has the parameter `process`.

Figure 47 shows an example of parameters for a Rule template.

Name	Type	Configuration
periodType	SelectConstant	"days": "days", "hours": "hours", "minutes": "minutes"
businessValue	InputInteger	
period	InputInteger	
priority	InputInteger	
process	SelectDynamic	evo.gtl.config.RuleSelectProcessSource
periodFrom	InputInteger	
periodTo	InputInteger	
customAttribute	InputText	
businessCalendar	SelectDynamic	evo.bc.config.RuleSelectBusinessCalendarSource
periodTypeBC	SelectConstant	"days": "working days", "hours": "working hours", "minutes": "working minutes"
operator	SelectConstant	"EQ": "equal to", "NE": "not equal to", "LT": "less than", "LE": "less than or equal to", "GT": "greater than"
department	SelectDynamic	evo.gtl.config.RuleSelectDepartmentSource
taskChannels	LookupTable	channels
skill	SelectDynamic	evo.gtl.openmedia.config.RuleSelectSkillSource

Figure 47: Rules Template—Parameters Tab

Parameters that are used as simple input controls appear only as specific names in action and condition definitions. If a parameter (such as a combo box) is more complex than a simple input control, it must be defined in the Parameters tab. Table 9 lists the properties in the Parameters tab.

Table 9: Parameters Tab Properties and Actions

Property/Action	Description
Parameter Name	The name of the parameter. This field is mandatory.
Parameter Type	<p>The type of input control for the parameter. There are six input control types:</p> <ul style="list-style-type: none"> • SelectConstant: constant combo box with labels and values. • SelectDynamic: dynamic combo box. Labels and values are received from the specific Java class that implements the interface <code>evo.rules.cmc.RuleSelectDynamicSource</code>. • InputText: input box that allows you to enter any text. • InputInteger: input box that allows you to enter integer values. • InputNumber: input box that allows you to enter number values. • InputDate: input box that allows you to enter date values. • LookupTable: drop-down box with key value pairs that were defined under Tenant > Lookup Tables. • SelectDynamicFiltered: dynamic combo box with a filter. That Java class that implements <code>RuleSelectDynamicFilteredSource</code>. <p>This field is mandatory.</p>
Configuration	<p>The attributes for SelectConstant and SelectDynamic type parameters.</p> <p>The pattern for SelectConstant is the following: "<code><value_1></code>": "<code><label_1></code>", "<code><value_2></code>": "<code><label_2></code>", ..., "<code><value_n></code>": "<code><label_n></code>"</p> <p>The pattern for SelectDynamic is the following: <code><package name>.<class name></code></p> <p>This field is mandatory for SelectConstant and SelectDynamic type parameters.</p>

Mapping several instances of a parameter to a single parameter definition

The underscore (`_`) character in parameter names has a special meaning when building rule templates. It is used to specify an index of the parameter, if the rule expression requires more than one instance of the particular parameter. The most common example is a range definition.

For example, suppose you need to create a condition which has to check if the task's due date is in the date1 to date2 range, or in the date3 to date4 range. There are two possible approaches:

1. Create a condition such as:
 Due is in "{dueDT1}" to "{dueDT2}" or in "{dueDT3}" to "{dueDT4}"
 This will require the definition of 4 parameters with type InputDate in the Parameters section. This approach is not very convenient, especially if there is more than one occurrence of the condition/action.
2. Use underscore and index:
 Due is in "{dueDT_1}" to "{dueDT_2}" or in "{dueDT_3}" to "{dueDT_4}"
 Using this approach, you need to specify only one parameter, with name dueDT and type InputDate.

Note: Rules will not be executed properly if you use underscores while defining parameters in the rule template (for example, dueDT or thisIsMyTextParam will work fine, whereas due_DT or this_Is_My_Test_Param will not be recognized). Use underscores only when referring to the multiple instances of the same parameter when creating conditions and actions, as described above.

Functions

When defining rule templates, elements that are more complex than conditions and actions might be required. The Functions tab (see [Figure 48](#)) enables you to write specific Java functions for different purposes for use in rule templates. The specified functions are used in the Rule Language Mappings.

Note: During use of a rule function in a condition or action, make sure that each function name and parameter is prefixed with a space, such as the following; otherwise, the function might not be recognized by the rules parser:

```
eval( someFunction( param1, param2) )
```

System > Rule Templates > Standard Rules Template

ID **Rule Template Name**

SRT Standard Rules Template

Conditions **Actions** **Parameters** **Functions**

Function Name	+	-
getCurrentDT	+	-
getPeriodDTFrom	+	-
setDepartmentAndProce	+	-
isProcess	+	-
setTime	+	-
getDueDT	+	-
compareInteger	+	-
setIntegerValue	+	-
setStringValue	+	-
increaseIntegerValue	+	-
multiplyIntegerValue	+	-

Name

getCurrentDT

Description

getCurrentDT

Body

```
import java.util.Date;
import java.util.Calendar;

function Date getCurrentDT()
{
    Calendar localCalendar = Calendar.getInstance();
    int offset = localCalendar.get(Calendar.ZONE_OFFSET) +
localCalendar.get(Calendar.DST_OFFSET);
    Date localDT = localCalendar.getTime();
    return new Date(localDT.getTime()-offset);
}
```

Save **Save & Close** **Cancel** **Delete**

Figure 48: Rules Template—Functions Tab

Table 10 lists the properties that are available in the Functions tab.

Table 10: Functions Tab Properties and Actions

Property/Action	Description
Function Name	The name of the function. This property is mandatory.
Description	A short description of the function. This property is optional.
Body	The function code, with the following syntax: function <return type> <function name>(<arguments>) {...} This field is mandatory.

Metrics Templates

Metrics templates can be used to define various types of metrics that will be available at the business user level to specify actual values for departments and processes. Metrics templates are just like any other module components and must be assigned to a module to be available under departments and processes.

Figure 49 shows an example.

The screenshot shows a configuration window titled "System > Metric templates > Financial Metrics". Inside the window, there is a "Name" field containing "Financial Metrics" and a "Description" text area. Below these is a section titled "Metric types" which contains a table. The table has four columns: "Name", "Type", "Distribution point", and "Description". One row is visible in the table with the following values: "Average Cost Per Task", "Number", an unchecked checkbox, and "Average Cost Per Task". At the bottom of the window, there are four buttons: "Save", "Save & Close", "Cancel", and "Delete".

Figure 49: MetricsTemplate

Table 11 lists value types for the Metric template. Refer to “Metrics” on page 203 for more information about Metrics.

Table 11: Metric Template Value Types

Value Type	Description
Time	Time value in days, hours, minutes, or hours.
Flag	Check box for true/false values.
Number	Number value, basic number validation.

Table 11: Metric Template Value Types (Continued)

Value Type	Description
Percentage	Percentage from 0% to 100%.
Date	Calendar control for selecting month, day, and year.
Lookup Table	Drop-down list of Lookup Tables configured under the Tenant.

Solutions and Services

The Services configuration section allows the setup of iWD runtime instances for a specific infrastructure and environment:

- Use Solution Details view (see [page 137](#)) to create new solutions, as well as to view and modify general solution attributes.
- Use Runtime Nodes view (see [page 138](#)) to define runtime nodes in which services will run.
- Use Services view (see [page 139](#)) to monitor and manage service status.
- Use Service Details view (see [page 140](#)) to create new services, as well as to view and modify configuration of existing services.
- Use Deployment view (see [page 143](#)) to deploy solution configuration to runtime nodes, as well as to activate services.
- Use Change History view (see [page 144](#)) to check the history of each deployed configuration version.

Note: It is recommended that you do **not** create any Solutions and Services under the System Tenant. You should do so under a Managed Tenant. The recommended role for the System Tenant is to be the place where you create all of your Modules and Components (which is done by importing the various configuration xml files); these Modules can then be inherited by the Managed Tenants and assigned or unassigned as needed at the Managed Tenant level.

Solution Details

Solution Details view displays, and lets you modify, general solution attributes. For more information on solutions, refer to “Solutions” on [page 30](#) of [Chapter 2](#). [Figure 50](#) shows an example.

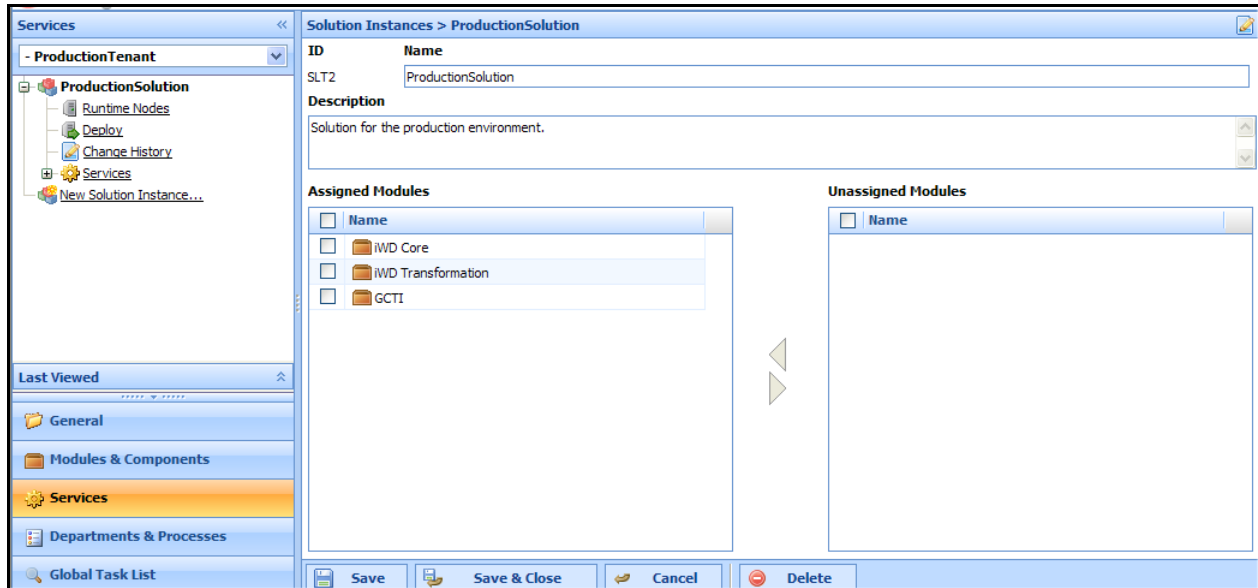


Figure 50: Solution Details

[Table 12](#) lists the properties and actions available in Solution Details view.

Table 12: Solution Details Attributes and Actions

Property/Action	Description
ID	The ID of the solution. This field becomes read-only after the solution is created.
Name	The name of the solution.
Description	A description of the solution.
Assigned Modules	Modules (or functionality) that are assigned to the solution. Components such as Services and Rule templates from these modules are available for use in the solution.
Unassigned Modules	Modules that have not been assigned to the solution. Components from these modules are not available for use in the solution.
Save/Save & Close/Cancel/Delete	Standard iWD Manager functions as described in “Object Actions” on page 108 .

Runtime Nodes

Runtime Nodes view lets you view and configure the runtime nodes in which the solution's services are running. See “Runtime Nodes” on [page 30](#) for more information about runtime nodes. [Figure 51](#) shows an example.

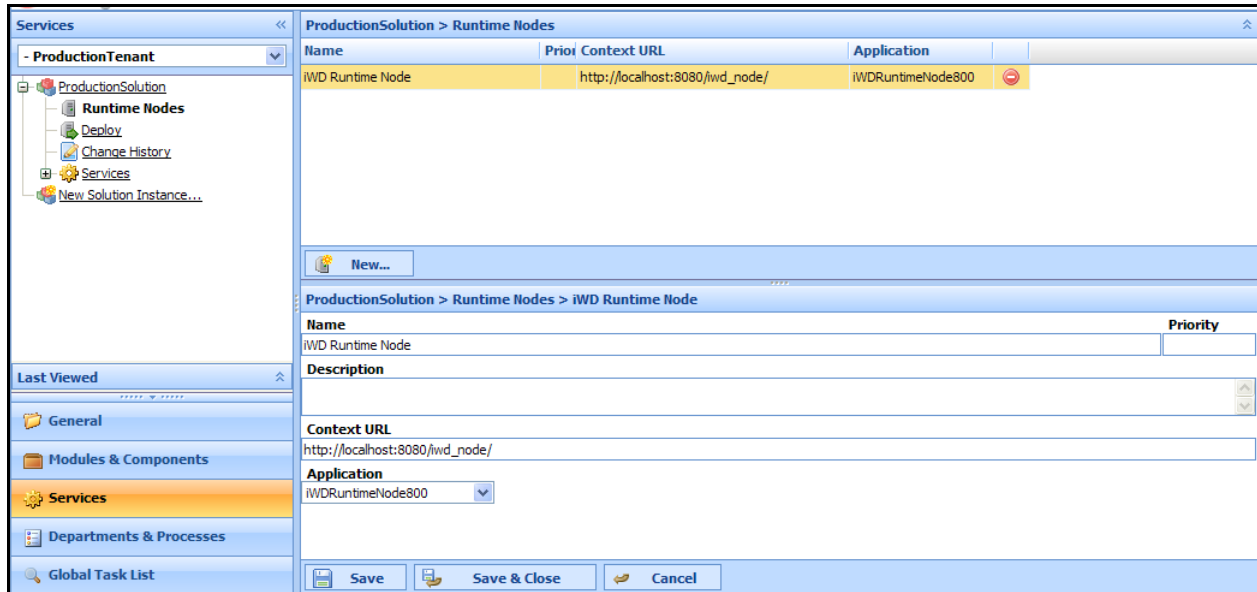


Figure 51: Runtime Node

[Table 13](#) describes the properties and actions that are available in Runtime Nodes view.

Table 13: Runtime Node Properties and Actions

Property/Action	Description
Name	The name of the runtime node.
Context URL	The URL of the runtime node. This depends on how the runtime node was installed. For example, a possible context URL is <code>http://localhost:8080/iwd_node/</code> , where <code>8080</code> is the port used by Tomcat, and <code>iwd_node</code> is the Runtime Node webapp in the Tomcat container.
New...	Allows for definition of a new runtime node.

Table 13: Runtime Node Properties and Actions (Continued)

Property/Action	Description
Application	The iWD Runtime Node application, as configured in the Configuration Server database. This application is the one that will manage the connection to Interaction Server. Each Runtime Node must have its own configured Runtime Node application in Configuration Manager/Genesys Administrator.
Save/Save & Close/Cancel/Delete	Standard iWD Manager functions as described in “Object Actions” on page 108 .

Services

Services view displays the status of each service that has been configured in the solution and allows you to manage service status manually. Refer to “Services” on [page 31](#) of [Chapter 2](#) for more information about services. [Figure 52](#) shows an example.



McrSolution1 > Services				
Runtime Node	Service Name	Status	Status Message	Log
Default	McrSolution1CfqSrvConnector	Started		View
Default	McrSolution1ESP1	Started		View
Default	McrSolution1InxSrvConnector1	Started		View
Default	McrSolution1Logging	Started		View
Default	McrSolution1Rules	Started		View
Default	McrSolution1XMLCP1	Started		View

Refresh

Figure 52: Services View

[Table 14](#) describes the properties and actions that are available in the Services view.

Table 14: Services View Properties and Actions

Attribute/Action	Description
Runtime Node	The runtime node on which the service is running. This field is absent for any service that runs on all available runtime nodes.
Service Name	The name of the service. Click here to open the Service Details view (see page 140).
	Starts a stopped service.
	Stops a started service.
Status	<p>Service status:</p> <ul style="list-style-type: none"> • Started—the service is active (running). • Scheduled—the service is scheduled to run in the future. • Queue—the service is scheduled to run after another job that is in the same execution queue has finished. • Stopped—the service is stopped. • Error—the service has encountered an error during its last operation; error details are provided in the Status Message column. • Not Deployed—the service is configured in the solution, but the configuration is not yet deployed to the runtime node. • Not Accessible—the runtime node is not accessible. This can be caused by an invalid runtime node configuration or because the application server on which the node is installed is not running.
Status Message	Displays additional service-status details, when available, such as an error message. Click it to view details in a larger window (shortcut menu).
Log	Opens the Log Viewer dialog box (see page 145) for the specified service.

Service Details

Service Details view (see [Figure 53](#)) displays, and lets you modify, service attributes. See also “iWD Services” on [page 146](#) for a detailed reference on specific service types.

McrSolution1 > Services > McrSolution1CfgSrvConnector

ID **Service Name**
 CfgSrvConnector McrSolution1CfgSrvConnector

Service Template **Runtime Node**
 Configuration Server Connector Default

Description

Properties

Name	Default	Value
startAutomatically		<input checked="" type="checkbox"/>
logLevel		Default
CMEApplicationName		Mcr800_CfgServerConnector
reconnectionAttempts	<input checked="" type="checkbox"/>	3
reconnectionPeriod	<input checked="" type="checkbox"/>	2
protocolTimeout	<input checked="" type="checkbox"/>	60000
eventBufferSize	<input checked="" type="checkbox"/>	5000

Save Save & Close Cancel Delete Copy

Figure 53: Service Details

Table 15 describes the properties and actions that are available in the Service Details view.

Table 15: Service Details View Properties and Actions

Property/Action	Description
ID	The ID of the service. This field becomes read-only after the service has been created. When a new service is created in iWD configuration, the default ID generated by iWD is based on the type of service being created. For example, if a new Web Service Capture Point service is created, the default ID generated by iWD will be WS_CP_x, where x is an integer that increments as additional Web Service Capture Point services are created.
Service Name	The name of the service.

Table 15: Service Details View Properties and Actions (Continued)

Property/Action	Description
Service Template	The template on which the service is based. This field becomes read-only when the service is created. See “Modules and Components” on page 125 for more information about service templates.
Runtime Node	The runtime node on which the service runs. This field is absent for any service that is running on all available runtime nodes.
Description	A description of the service.
Properties	<p>The properties of the service that configure its functionality:</p> <ul style="list-style-type: none"> • Name—the name of the property. • Default—whether to use the property’s default value. • Value—the property’s value. Depending on property type, this field can be a text box, check box, or dropdown box. When the Default box has been checked, this field is read-only. <p>Different service types have different properties. For a detailed reference on specific service types, see “iWD Services” on page 146.</p>
Save/Save & Close/Cancel/Delete	Standard iWD Manager functions as described in “Object Actions” on page 108 .
View Tasks	Opens the Global Task List view in iWD Manager. This action is available only for capture points.

Deployment

Changes to the configuration in iWD Manager are not automatically activated. Deploy, shown in [Figure 54](#), performs this task by distributing a solution's configuration across the defined runtime nodes.

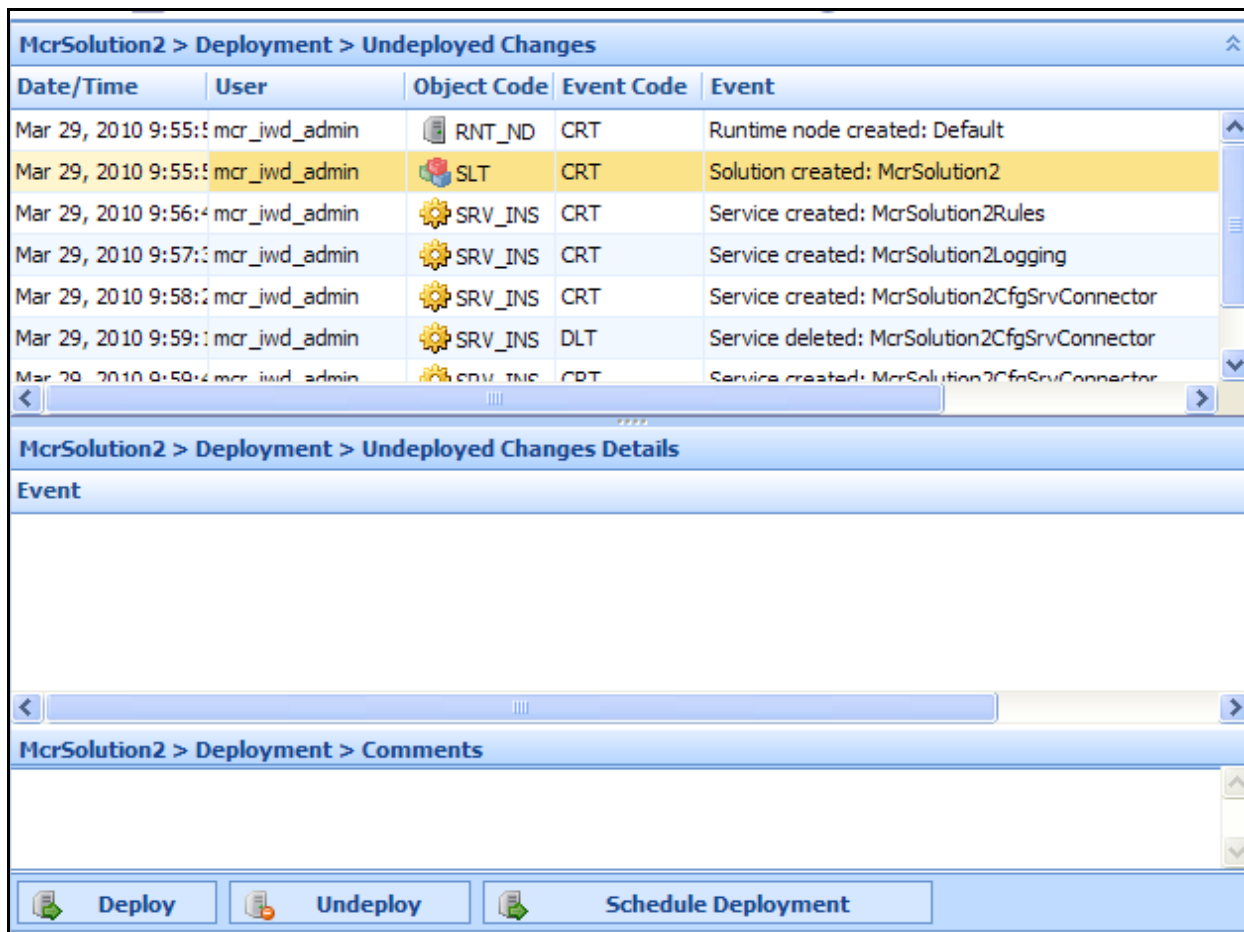


Figure 54: Deploy

[Table 16](#) describes the properties and actions that are available in Deploy.

Table 16: Deploy Properties and Actions

Property/Action	Description
Undeployed Changes/ Undeployed Changes Details	Detailed information about activities performed in iWD Manager since the last deployment. For more details, see “History” on page 123 .
Comments	Deployment comments. They will be displayed as version comments in the Change History (see page 144).

Table 16: Deploy Properties and Actions (Continued)

Property/Action	Description
Deploy	<p>Deploys the configuration. Depending on the configuration's complexity and runtime environment characteristics, this action can take several seconds.</p> <p>Note: Deployment deploys the entire solution whether it is done from the Services section or from the Departments and Processes section. When deploying changes in the Departments and Processes section, all changes made to Services will also be deployed.</p> <p>All configured runtime nodes must be available to deploy. Deployment generates and copies a set of configuration XML files and JBoss rules files to runtime nodes. The configuration files are stored in the WEB-INF/config directory under the runtime node, but the Deploy function handles the files and manual intervention or manipulation of the files is not needed. All files are always regenerated and overwritten each time the configuration is deployed.</p>
Undeploy	<p>Stops all of the services of a solution, and undeploys the configuration from all available runtime nodes. If any runtime node that is defined in the solution is not available, iWD Manager will display a warning message.</p> <p>All configured services are deactivated. Deployed configuration files are removed and the runtime node reloads the empty configuration.</p>
Schedule Deployment	<p>Allows deployment to be performed on a specific date/hour. This allows you to deploy changes during a quiet period, when there is nothing happening in the system.</p> <p>Note: If you have more than one iWD Manager instance connected to a single iWD Configuration database, there <code>wd.host</code> parameter in the <code>wd.properties</code> file must be unique for every host on which iWD Manager is running. The <code>wd.properties</code> file can be found in the web application server's <code>webapps</code> folder for iWD Manager. For example, for Tomcat this might be <code><TomcatDirectory>\webapps\iwd_manager\WEB-INF\classes</code></p>

Change History

Change History maintains the history of deployed configuration versions. Whenever a solution configuration is deployed, a new configuration version

record is created that contains all of the changes that have been made in the configuration since the previous version (deployment). [Figure 55](#) shows an example.

McSolution1 > Change History > Versions

Deployed On	Deployed By	Scope	Comments
Apr 13, 2010 4:53 PM	default	SOLUTION	
Apr 13, 2010 5:58 AM	mcr_iwd_admin	SOLUTION	

Context History

Date/Time	User	Object Code	Event Code	Event
Apr 13, 2010 4:53:0	default	SRV_INS	UPD	Service updated: McSolution1Logging
Apr 13, 2010 4:52:4	default	SRV_INS	UPD	Service updated: Logging
Apr 13, 2010 4:51:3	default	RL	CRT	Rule created: McSolution1_Deprt1_Proc1_Rule1, assoc
Apr 13, 2010 4:47:0	default	RL	CRT	Rule created: McSolution1_Deprt1_Rule1, associated t
Apr 13, 2010 4:45:1	default	RL	CRT	Rule created: McSolution1_GRule1, associated to: Mcr

Details

Figure 55: Change History

[Table 17](#) describes the attributes and actions that are available in Change History view.

Table 17: Change History Attributes and Actions

Attribute/Action	Description
Deployed On	When the configuration was deployed.
Deployed By	Who deployed the configuration.
Comments	Deployment comments.

Log Viewer

Log Viewer (see [Figure 56](#)) allows you to view iWD service log files remotely. Various viewing methods can be used, including near-real-time viewing with

Auto Refresh functionality, as well as viewing of historical log files for the particular service.

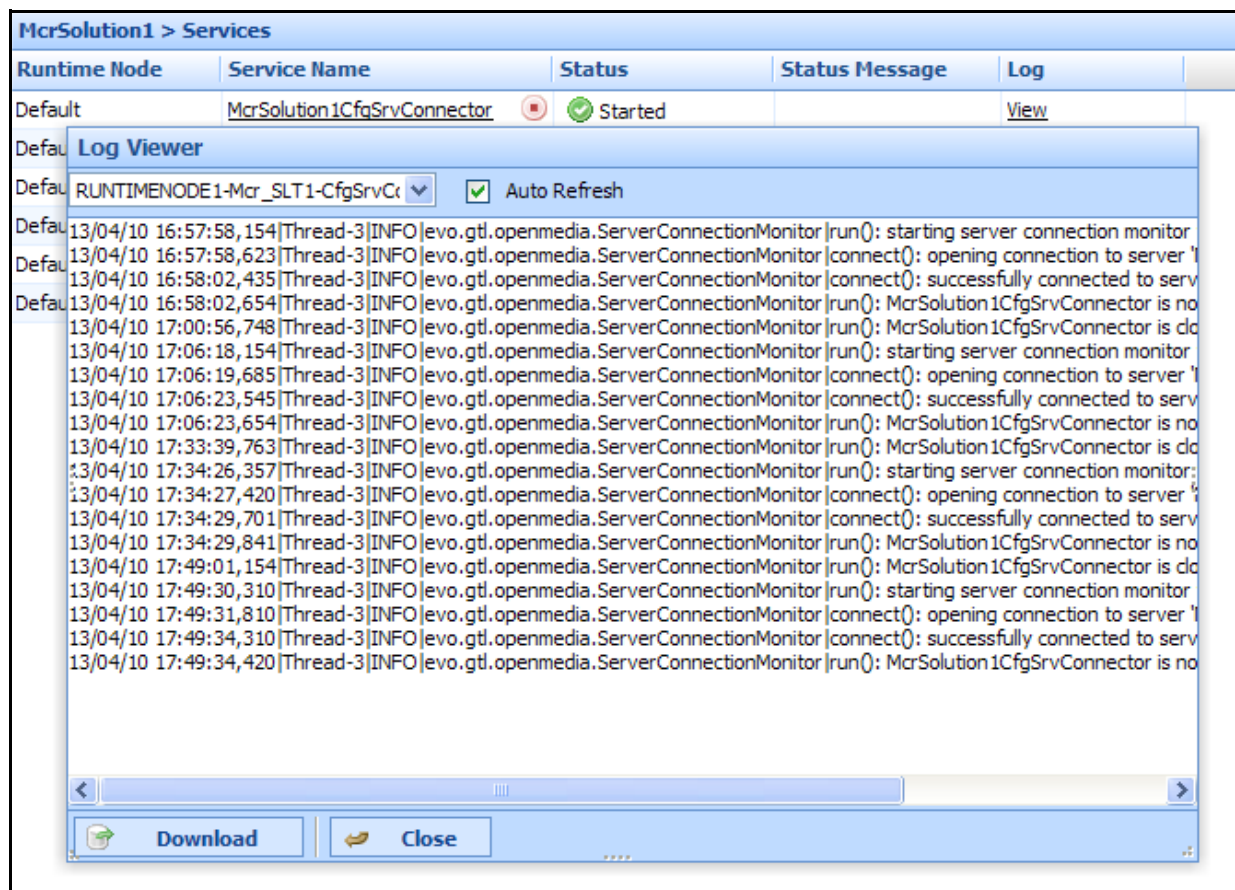


Figure 56: Log Viewer

iWD Services

iWD services implement actual iWD functionality, such as capturing tasks or loading data into the Data Mart. iWD Services are classified as Core Services, Capture Points, or Reporting Services.

Note: Services are configured per solution, and each solution must be configured under a tenant in iWD Manager. Therefore, before configuring services, you must configure a tenant and solution. Refer to Chapter 3, “Installation,” on [page 37](#) for more information.

Core iWD Services

Core iWD services are mandatory in any iWD solution and enable the core iWD functionality:

- **Rules Service**—Is a mediator between iWD services and rule engine implementation. It allows iWD services to execute various rules
- **Logging Service**—Provides internal logging capabilities within an iWD solution
- **Configuration Server Connector Service**—Maintains a physical connection to the Genesys Configuration Server
- **Interaction Server Connector Service**—Maintains a physical connection to the Genesys Interaction Server
- **Business Context Management Service**—Works in conjunction with the Rules service to enable business rules to be invoked from within routing strategies.

Capture Points

Capture points represent an interface for feeding tasks to iWD. A number of different capture points are available, such as the following:

- **Web Service Capture Point Service**—Provides the ability for third-party applications to submit tasks to iWD via a Web service interface. It also provides an interface for managing submitted tasks, such as holding and restarting.
- **Database Capture Point Service**—Captures tasks directly from database tables. Configuration for this is based on SQL.
- **XML File Capture Point Service**—Captures tasks from XML files. The files should comply with the iWD Task XML schema.
- **WebSphere MQ Capture Point Service**—Captures tasks from systems using IBM WebSphere MQ as a message bus. Refer to the *iWD 8.0 WebSphere MQ Capture Adapter Reference Guide* for more information about this capture point.
- Beginning in version 8.0.2, eServices offers integrated **JMS** Capture Point functionality. Refer to the eServices documentation for more information. Version 8.0.210 of eServices (specifically, the Interaction Server component) also includes integrated XML File Capture Point functionality. Version 8.1.0 of eServices (the Interaction Server component) includes an integrated Database Capture Point.

Note: In order for iWD to function properly, a Capture Point ID is required for all tasks. Therefore, when capturing tasks through media servers other than iWD capture points, such as through Genesys E-mail Server or Genesys SMS Server, you must configure a “dummy” capture point. Refer to Appendix C, “Adapting the iWD Business Process for Standard Genesys Channels,” on [page 275](#) for additional information.

Reporting

The Statistics Adapter Service allows performance of custom aggregations on data and sends the resulting statistics to Genesys Stat Server. Statistics then can be viewed in CCPulse+ or any other Stat Server client.

Historical reporting is enabled by a number of ETL jobs that transform and load iWD runtime data into a separate reporting database that is called the iWD Data Mart (see [Table 18](#) for a list of preconfigured ETL jobs). Essentially, the iWD Data Mart is a set of star schemas that contain historical iWD data that is optimized for reporting. The ETL jobs are set up as scheduled services in iWD. Refer to the *iWD Data Mart Reference Guide* for more information.

Table 18: Preconfigured ETL Jobs

Job Function	Attribute	Description
Initialize iWD Data	Job Name	Initialize
	Function	Initializes the necessary data structures, and populates static dimensions, such as the AGE dimension.
	Schedule	Runs once.
Load Configuration	Job Name	Load Configuration
	Function	Loads updates from iWD configuration tables into dimension tables.
	Schedule	Configurable through service properties; typically runs on a 15-minute cycle.
Load Intraday	Job Name	Load Intraday
	Function	Loads updates from the Interaction Server database tables into core fact tables.
	Schedule	Configurable through services properties; recommended that it be scheduled to run after the Load Configuration Job ends through the Job Dependency option.

Table 18: Preconfigured ETL Jobs (Continued)

Job Function	Attribute	Description
Aggregate Intraday	Job Name	Aggregate Intraday
	Function	Aggregates data that previously was loaded into fact tables by the Load Intraday Job into aggregation tables.
	Schedule	Recommended that it be scheduled immediately after the Load Intraday Job has completed—typically, running every 15 minutes. The frequency of this aggregate job does not have any bearing on the 15-minute aggregate that is being populated.
Aggregate Statistics	Job Name	Aggregate Statistics
	Function	Generate statistics by executing statistics plug-ins.
	Schedule	Recommended that it be scheduled immediately after the Aggregate Intraday Job has completed, because most of the statistics plug-ins are using aggregated facts.
Load Historical	Job Name	Load Historical
	Function	Moves data from intraday fact tables into historical fact tables.
	Schedule	Runs daily through the schedule that is defined in the service properties.
Aggregate Historical	Job Name	Aggregate Historical
	Function	Aggregates data from historical fact tables into 15-minute, hourly, daily, and monthly aggregation tables.
	Schedule	Runs once a day, after the Load Historical Job.
Maintain iWD	Job Name	Maintain
	Function	Cleanses historical tables—that is, removes expired facts from these tables, based on rules that are defined on the ETL Service property in iWD Manager.
	Schedule	Runs once a day, after the Aggregate Historical Job.

Table 18: Preconfigured ETL Jobs (Continued)

Job Function	Attribute	Description
Prune events and interactions	Job Name	Prune
	Function	Removes event logs and interactions from the databases that are loaded in the Data Mart when tasks/interactions have reached their final state.
	Schedule	Runs once a day, after the Maintain Job.

Workforce Management

iWD 8.0 supports Workforce Management objects. A new service has been added, that uses the template “Workforce Management Connector”.

Configuring this service allows you to see some objects that exist in the WFM database.

Recommended Order of Configuration for iWD Services

[Table 19](#) lists the iWD services in their recommended order of configuration and indicates any dependencies on other services.

Note: The following table includes all iWD services. The Installation chapter does not provide information on how to configure each service, only some of the services. The general procedure is the same for each service.

Table 19: iWD Services

Name	Mandatory	Category	Dependencies	Notes
Logging Service	Yes	Core	None	See page 152 .
Configuration Server Connector	Yes	Core	None	See page 153 .
Interaction Server Connector	Yes	Core	Genesys Configuration Server Connector	See page 154 .
Scripting Service	No	Core	None	Required only when iWD transformation is being used in WebSphere MQ or XML Capture Points. See page 156 .

Table 19: iWD Services (Continued)

Name	Mandatory	Category	Dependencies	Notes
Web Service Capture Point	No	Capture Point	None	See page 157 .
XML File Capture Point	No	Capture Point	Scripting Service*	See page 159 . *Scripting Service is only required when iWD transformation is being used.
Database Capture Point	No	Capture Point	None	See page 161 .
WebSphere MQ Capture Point	No	Capture Point	Scripting Service*	See the <i>iWD 8.0 WebSphere MQ Capture Adapter Reference Guide</i> for information about deploying this component. *Scripting Service is only required when iWD transformation is being used.
Rules Service	Yes	Core	None	See page 167 .
Business Context Management Service	Yes	Core	Genesys Configuration Server Connector	See page 168 .
Database Service	No	Reporting	None	See page 169 .
Statistics Adapter	No	Reporting	None	See page 170 .
Kettle ETL Service	No	Reporting	Database	See page 173 .
Scheduled ETL Job	No	Reporting	Kettle ETL Service	A separate scheduled ETL job that must be created for each job type. See page 176 .
WFM Connector	No	Workforce Management	Genesys Configuration Server Connector	See page 180 .

Service Log Levels

For each service, a log level can be specified in the `logLevel` property. The level should be set to `Default` unless otherwise directed by Genesys Technical Support.

The possible log levels are:

- `Default`—uses the log level that was configured for the Logging Service (see “Logging Service” on [page 152](#)).
- `Debug`—the most detailed informational events that are most useful in debugging an application.
- `Info`—informational messages that highlight the progress of the application.
- `Warning`—potentially harmful situations.
- `Error`—error events that might not affect the application’s ability to run.
- `Trace`—turns on all logging.
- `Off`—turns off all logging.

Logging Service

The Logging Service is a mandatory iWD service that provides internal logging capabilities within an iWD solution.

Note: The Logging Service must be configured even if you only want to create logs through the Genesys Message Server.

In addition to the options and actions described in “Service Details” on [page 140](#), the properties listed in [Table 20](#) are configurable for the Logging Service.

Table 20: Logging Service Properties

Property	Description
<code>startAutomatically</code>	Indicates whether the service should be started automatically after the configuration deployment.
<code>logLevel</code>	The Service log level. This should be set to <code>Info</code> unless otherwise instructed by Genesys Technical Support. See “Service Log Levels” on page 152 for a description of each log level.
<code>logAgeInDays</code>	Sets the number of days that log files should be kept in the system. A value of <code>0</code> disables this limit. This property is not available if <code>logLevel</code> is set to <code>Off</code> .

Table 20: Logging Service Properties (Continued)

Property	Description
logMaxFileSizeMegaBytes	Sets a limit on the size of a single log file, in megabytes. A value of 0 disables this limit. This property is not available if logLevel is set to Off.
logFilesToKeep	Sets a limit on the number of log files that are kept for this service, excluding the current log file. A value of 0 disables this limit. This property is not available if logLevel is set to Off.
logDirectory	<p>The directory in which the log files will be stored, for all services. If it starts with / (on Unix-based operating systems) or a drive letter (on Windows), an absolute path will be used; otherwise, the path is relative from the web applications folder on the application server.</p> <p>Note: It is strongly recommended that you only set the file path to a directory on a local machine, not a remote location such as a shared network drive. Logging to a remote location can severely impact performance.</p>

Configuration Server Connector Service

The Configuration Server Connector service maintains a physical connection to the Genesys Configuration Server. The Configuration Server Connector *service* needs to be created manually in iWD Manager.

In addition to the options and actions described in “Service Details” on [page 140](#), the properties listed in [Table 21](#) are configurable for the Configuration Server Connector service.

Table 21: Configuration Server Connector 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 otherwise instructed by Genesys Technical Support. See “Service Log Levels” on page 152 for a description of each log level. Depending on the setting of this property, additional logging properties might be available. Refer to “Logging Service” on page 152 for descriptions of these common properties.
reconnectionAttempts	<p>Exception handling:</p> <p>The maximum number of attempts to connect to Configuration Server, before connecting to the backup server.</p>

Table 21: Configuration Server Connector Service Properties (Continued)

Property	Description
reconnectionPeriod	Exception handling: The time, in seconds, between individual reconnection attempts.
protocolTimeout	Exception handling: The timeout of the Genesys Configuration Server protocol; specifies the number of milliseconds that the connector will wait for Configuration Server to respond to the request.
eventBufferSize	Performance tuning: The number of Configuration Server events that can be queued up in memory, until they are processed by the service.
addpTimeout	Used to configure local and remote timeouts (respectively) for Advanced Disconnect Detection Protocol (ADDP). In Configuration Manager or Genesys Administrator, you should enable ADDP on the connection to Interaction Server on the Connections tab of your iWD Runtime Node application.
addpRemoteTimeout	Used to configure local and remote timeouts (respectively) for Advanced Disconnect Detection Protocol (ADDP). In Configuration Manager or Genesys Administrator, you should enable ADDP on the connection to Interaction Server on the Connections tab of your iWD Runtime Node application.

Interaction Server Connector Service

The Interaction Server Connector service maintains a physical connection to the Genesys Interaction Server. This service uses the Genesys Configuration Server Connector service to retrieve additional configuration data, such as the name of the host and the port of the target Interaction Server.

In addition to the options and actions described in “Service Details” on [page 140](#), the properties listed in [Table 22](#) are configurable for the Interaction Server Connector service.

Table 22: Interaction Server Connector 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 otherwise instructed by Genesys Technical Support. See “Service Log Levels” on page 152 for a description of each log level. Depending on the setting of this property, additional logging properties might be available. Refer to “Logging Service” on page 152 for descriptions of these common properties.
protocolTimeout	Exception handling: The timeout of the Interaction Server protocol; specifies the number of milliseconds that the connector will wait for Interaction Server to respond to the request.
eventBufferSize	Performance tuning: The number of Interaction Server events that can be queued up in memory, until they are processed by the service.
threads	Performance tuning: The size of the thread pool.
attributeFilterInclude	This value of this property is used to determine which interaction attached data keys will be used to trigger taskUpdated notifications to a source system, and which interaction attached data keys will be shown on the iWD Manager's Attributes tab when the task is selected in the Global Task List. A subset of what is defined by this filter may be negated by the filter specified in the attributeFilterExclude property. The wildcards that are allowed are the following: * (asterisk) any number of any characters ? (question mark) any character. For example: *Agent*: Any pair whose name contains “Agent” ?code: Any pair whose name starts with any character and ends with “code”

Table 22: Interaction Server Connector Service Properties (Continued)

Property	Description
attributeFilterExclude	<p>This value of this property is used to determine which interaction attached data keys will be filtered out of the list that will trigger taskUpdated notifications to a source system, and which will be shown on the iWD Manager's Attributes tab, when the task is selected in the Global Task List. Any attached data key matching those listed in the attributeFilterExclude property will be filtered out. By default this filter includes the URS-specific attached data keys.</p> <p>The wildcards that are allowed are the following:</p> <ul style="list-style-type: none"> * (asterisk) any number of any characters ? (question mark) any character <p>For example:</p> <ul style="list-style-type: none"> *Agent*: Any pair whose name contains "Agent" ?code: Any pair whose name starts with any character and ends with "code" <p>Note: It is important to configure this property correctly if you are using any custom task attributes so that they will be displayed on the Attributes tab of the Global Task List as expected.</p>
configurationServerConnector	Mandatory dependency: The Configuration Server Connector service (see page 168); provides access to the Configuration Server.
addpTimeout	Used to configure local and remote timeouts (respectively) for Advanced Disconnect Detection Protocol (ADDP). In Configuration Manager or Genesys Administrator, you should enable ADDP on the connection to Interaction Server on the Connections tab of your iWD Runtime Node application.
addpRemoteTimeout	Used to configure local and remote timeouts (respectively) for Advanced Disconnect Detection Protocol (ADDP). In Configuration Manager or Genesys Administrator, you should enable ADDP on the connection to Interaction Server on the Connections tab of your iWD Runtime Node application.

Scripting Service

The Scripting Service is an optional iWD service that provides scripting capabilities for message transformation purposes for XML File Capture Points

(see [page 159](#)) and WebSphere MQ Capture Points (see the *iWD 8.0 WebSphere MQ Capture Adapter Reference Guide* for more information).

Note: The iWD Scripting Service is not used for the JMS Integrated Capture Point. Groovy scripting for XML transformation is supported for the JMS Capture Point, but the scripting is built into Interaction Server. Refer to the *eServices 8.0 User's Guide* for more information.

In addition to the options and actions described in “Service Details” on [page 140](#), the properties listed in [Table 23](#) are configurable for the Scripting Service.

Table 23: Scripting 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 otherwise instructed by Genesys Technical Support. See “Service Log Levels” on page 152 for a description of each log level. Depending on the setting of this property, additional logging properties might be available. Refer to “Logging Service” on page 152 for descriptions of these common properties.

Web Service Capture Point

The Web Service Capture Point is an iWD service that allows third-party systems to submit and manipulate tasks in iWD via a SOAP Web Service interface.

Note: When developing an integration to a source system using the iWD Web Service Capture Adapter in a .NET environment, .NET Framework 2.0 or higher must be in place.

For more information on the Web Service Capture Point, see the iWD Web Service Capture API documentation.

Note: The iWD Web Service Capture API documentation is available at <https://sites.google.com/a/iwdlab.com/iwd8/capture/webservice>

In addition to the options and actions described in “Service Details” on [page 140](#), the properties listed in [Table 24](#) are configurable for the Web Service Capture Point.

Table 24: Web Service Capture Point 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 otherwise instructed by Genesys Technical Support. See “Service Log Levels” on page 152 for a description of each log level. Depending on the setting of this property, additional logging properties might be available. Refer to “Logging Service” on page 152 for descriptions of these common properties.
checkIfAlreadyCaptured	If true, iWD will verify whether another task that has a given captureId has already been captured.
timezone	The time zone of the Web Service Capture Point. Date/time values will be converted from the specified time zone to UTC, before those values are stored in the iWD. If this parameter is not specified, it defaults to the tenant time zone.
defaultMediaType	The media type attribute that will be assigned to tasks captured by the capture point. The list of available media types is retrieved dynamically from Configuration Server. In Configuration Server itself, the media type attribute is configured as a type of Business Attribute. The default media type used by the capture point may be overwritten if it is specified in the <mediaType> attribute in the <CreateTask> message.
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.
interactionServerConnector	Mandatory dependency: The Interaction Server Connector service (see page 154) that should be used for connectivity to the Interaction Server.
webserviceURLMapping	The webservice URL mapping. The URL is composed as follows: <runtime node context URL>/services/<webserviceURLMapping>, for example: http://server:8080/iwdnode/services/webserviceCapturePoint. To retrieve a WSDL file for the Webservice, attach ?WSDL to the URL.

Table 24: Web Service Capture Point Properties (Continued)

Property	Description
stringOnlyParameters	If selected, only string parameters will be accepted.
timeZoneIsSupplied	<p>If this property is selected (<code>timeZoneIsSupplied = ON</code>), then:</p> <ul style="list-style-type: none"> If the datetime has a timezone set, the date is treated as submitted in that timezone. If the datetime has no timezone set, the date is treated as submitted in the capture point's timezone. <p>The time format includes the timezone designator and the time without milliseconds, separated by a T: (<code>yyyy-MM-dd'T'HH:mm:ssZZ</code>) .</p> <p>For example: <code>2008-07-21T21:00:00+01:00</code></p> <p>If this property is not selected (<code>timeZoneIsSupplied = OFF</code>), then:</p> <ul style="list-style-type: none"> If the timezone is present in the datetime string, it is ignored, and the datetime is treated as submitted in capture point's timezone

XML File Capture Point

The XML File Capture Point is an iWD service for capturing tasks from XML files. The XML files can be formatted according to the standard iWD XML message schema (described in detail in “iWD Messages” on [page 225](#)) or in a custom XML format.

To handle custom XML formats, two transformation scripts must be created: one for input transformation, and one for output. iWD currently supports the Groovy scripting language 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. These transformation scripts are enabled if you import the `iwd_transform.xml` file into the iWD System or managed tenant. The `iwd_transform.xml` file is installed when you install the iWD Manager installation package, in the `config` directory. For example, at `C:\Program Files\GCTI\iWD Manager\config`.

In addition to the options and actions described in “Service Details” on [page 140](#), the properties listed in [Table 25](#) are configurable for the XML File Capture Point.

Table 25: XML File Capture Point Properties

Properties	Description
startAutomatically	Indicates whether the service should be started automatically after the configuration deployment.
logLevel	The Service log level. This should be set to <code>Default</code> unless otherwise instructed by Genesys Technical Support. See “Service Log Levels” on page 152 for a description of each log level. Depending on the setting of this property, additional logging properties might be available. Refer to “Logging Service” on page 152 for descriptions of these common properties.
checkIfAlreadyCaptured	If true, iWD will verify whether another task that has a given <code>captureId</code> already has been captured.
timezone	The time zone of the XML File 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 XML files will be converted to the specified time zone. If this parameter is not specified, it defaults to the tenant time zone.
defaultMediaType	The media type attribute that will be assigned to tasks captured by the capture point. The list of available media types is retrieved dynamically from Configuration Server. In Configuration Server itself, the media type attribute is configured as a type of Business Attribute. The default media type used by the capture point may be overwritten if it is specified in the <code><mediaType></code> attribute in the <code><CreateTask></code> message.
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 <code>expirationDateTime</code> attribute in the capture data.
interactionServerConnector	Mandatory dependency: The Interaction Server Connector service (see page 154) that should be used for connectivity to the Interaction Server.
threads	Performance tuning: The size of the thread pool.
idleSleepTimeSeconds	Service-idle period when there are no more tasks to process.
emergencySleepThreshold	Exception handling: Once distribution attempts fail the specified number of times, processing will be held.

Table 25: XML File Capture Point Properties (Continued)

Properties	Description
emergencySleepSeconds	Exception handling: Specifies how long to hold processing after emergencySleepThreshold failed distribution attempts.
incomingFileDirectory	The directory in which the XML File Capture Point looks for new files to capture.
capturedFileDirectory	The directory into which captured files are put.
completedFileDirectory	The directory into which completed tasks are exported. Each task is exported to an xml file with the naming convention <interaction id>.xml.
rejectedFileDirectory	The directory into which rejected tasks are exported. Each task is exported to an xml file with the naming convention <interaction id>.xml.
errorFileDirectory	The directory into which files that contain errors are placed.
TransformScriptingService	Optional dependency: Scripting Service (see page 167). If specified, XML input/output will be transformed using input/output transformation scripts.
TransformInputScript	The input transformation script; must be specified when TransformScriptingService is selected.
TransformOutputScript	The output transformation script; must be specified when TransformScriptingService is selected.
TransformOutputRootNode	The name of the XML root node in output files.
backupFor	High Availability: The primary XML File Capture Point in a high availability setup scenario. This service will perform a backup role in such a setup, and will take over processing if the primary service becomes unavailable.

Database Capture Point

The Database Capture Point is an asynchronous capture point service that creates tasks in iWD, based on records in a database. As a task in iWD goes through its life cycle stages, the Database Capture Point allows you to make updates to the database, to mimic the task life cycle in the originating system. The Database Capture Point allows execution of a certain database query or update statement, depending on the task state:

- **Capture**—the Database Capture Point scans the result set that is returned from the originating database by a special “capture” query and creates tasks in iWD that are based on the information that is provided by each result-set record.
- **Captured**—after the task is created in iWD, the Database Capture Point may execute a “captured” update query to update the respective database record and notify the originating system that the task for the respective row in the database has already been created.
- **Assigned**—the task has been assigned to a user. The Database Capture Point may execute an “assigned” update query to update the respective database record and notify the originating system that the iWD task for the respective row in the database has been assigned to the user.
- **Completed**—the task has been completed by a user. The Database Capture Point may execute a “completed” update query to update the respective database record and notify the originating system that the iWD task for the respective row in the database has been completed.
- **Restarted**—the task has been restarted (that is, returned to the iWD for reprocessing and distribution at a later time). The Database Capture Point may execute a “restarted” update query to update the respective database record and notify the originating system that the iWD task for the respective row in the database has been restarted.

In addition to the options and actions described in “Service Details” on [page 140](#), the properties listed in [Table 26](#) are configurable for the Database Capture Point.

Table 26: Database Capture Point 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 <code>Default</code> unless otherwise instructed by Genesys Technical Support. See “Service Log Levels” on page 152 for a description of each log level. Depending on the setting of this property, additional logging properties might be available. Refer to “Logging Service” on page 152 for descriptions of these common properties.
checkIfAlreadyCaptured	If true, iWD will verify whether another task that has a given <code>captureId</code> already has been captured.
timezone	The time zone of the target database. Date/time values will be converted from the specified time zone to UTC, before those values are stored in the iWD. If this parameter is not specified, it defaults to the tenant time zone.

Table 26: Database Capture Point Properties (Continued)

Property	Description
defaultMediaType	<p>The media type attribute that will be assigned to tasks captured by the capture point.</p> <p>The list of available media types is retrieved dynamically from Configuration Server. In Configuration Server itself, the media type attribute is configured as a type of Business Attribute.</p> <p>The core iWD attribute <code>mediaType</code> may be specified in the <code>CreateTask</code> message. If the <code>mediaType</code> attribute is returned as a result of the query configured in <code>captureQuerySQL</code>, then it will override the default <code>mediaType</code> specified in the Database Capture Point service property <code>defaultMediaType</code>.</p>
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 <code>expirationDateTime</code> attribute in the capture data.
interactionServerConnector	Mandatory dependency: The Interaction Server Connector service (see page 154) that should be used for connectivity to the Interaction Server.
threads	Performance tuning: The size of the thread pool.
idleSleepTimeSeconds	Service-idle period when there are no more tasks to process.
emergencySleepThreshold	Exception handling: Once distribution attempts fail the specified number of times, processing will be held.
emergencySleepSeconds	Exception handling: Specifies how long to hold processing after <code>emergencySleepThreshold</code> failed distribution attempts.
jdbcDriver	The name of the JDBC driver class to access the database.
jdbcURL	The URL of your JDBC connection. For information on how to construct a valid URL, see your JDBC driver documentation.
username	The name of the database user.
password	The password of the database user.
idField	The name of the column that uniquely identifies the record in the result set that is returned by the database query that is specified in the <code>captureQuerySql</code> parameter. The value of the field is mapped to the <code>captureId</code> parameter in the consecutive update statements.
processIdField	The name of the column that contains the process ID. If the value of this field matches a process ID in iWD, a process and department will be assigned to the task.

Table 26: Database Capture Point Properties (Continued)

Property	Description
fieldsToAttach	<p>A list of fields that will be attached to task data. Multiple fields should be separated by a comma. This property also supports * and ? wildcards.</p> <p>For example: <code>field1, field2, special*</code></p> <p>This expression means that the fields that have the names <code>field1</code> and <code>field2</code> will be attached, as well as any field that has a name that starts with <code>special</code>.</p>
captureQuerySql	<p>The database query that returns the result set in which each row will be captured as a task in iWD. The result set that is returned by this query must contain also the column that is specified in the <code>idField</code> parameter.</p> <p>For example: <code>select * from TABLE where STATUS="new"</code></p>
capturedUpdateSql	<p>The database update statement that updates the database to reflect that certain data has already been captured as a task in iWD. The <code>captureId</code> parameter can be used to reference the particular row.</p> <p>For example: <code>update TABLE set STATUS='captured' where idField=:captureId</code></p>
errorUpdateSql	<p>The database update statement that updates the database to reflect that the associated task in iWD has been held by an error. The <code>captureId</code> parameter can be used to reference the particular row.</p> <p>For example: <code>update TABLE set STATUS='error' where idField=:captureId</code></p>
assignedUpdateSql	<p>The database update statement that updates the database to reflect that the associated task in iWD has been assigned. The <code>captureId</code> parameter can be used to reference the particular row. The <code>userId</code> parameter can be used to set the name of the user who is working on the task.</p> <p>For example: <code>update TABLE set STATUS='assigned', USER=:userId where idField=:captureId</code></p>
completedUpdateSql	<p>The database update statement that updates the database to reflect that the associated task in iWD has been completed. The <code>captureId</code> parameter can be used to reference the particular row.</p> <p>For example: <code>update TABLE set STATUS='completed' where idField=:captureId</code></p>

Table 26: Database Capture Point Properties (Continued)

Property	Description
canceledUpdateSql	<p>The database update statement that updates the source database to reflect that the associated task in iWD has been canceled. The <code>captureId</code> parameter can be used to reference the particular row.</p> <p>For example:</p> <pre>update TABLE set STATUS='canceled' where idField=:captureId</pre>
heldUpdateSql	<p>The database update statement that updates the source database to reflect that the associated task in iWD has been put on hold. The <code>captureId</code> parameter can be used to reference the particular row.</p> <p>For example:</p> <pre>update TABLE set STATUS='held' where idField=:captureId</pre>
queuedUpdateSql	<p>The database update statement that will be executed when a task, captured by the Database Capture Point, is queued.</p> <p>For example:</p> <pre>update TABLE set STATUS='queued' where idField=:captureId</pre>
errorHeldUpdateSql	<p>The database update statement that updates the source database to reflect that the associated task in iWD has been put in an error held status due to an internal processing error. For example, a task may be put into an error held status if no Process is assigned to the task during the Classification phase of the rule execution. The <code>captureId</code> parameter can be used to reference the particular row.</p> <p>For example:</p> <pre>update TABLE set STATUS='errorHeld' where idField=:captureId</pre>
rejectedUpdateSql	<p>The database update statement that will be executed when a task, captured by the Database Capture Point, is rejected.</p> <p>For example:</p> <pre>update TABLE set STATUS='rejected' where idField=:captureId</pre>
restartedUpdateSql	<p>The database update statement that updates the database to reflect that the associated task in iWD has been restarted in iWD. The <code>captureId</code> parameter can be used to reference the particular row.</p> <p>For example:</p> <pre>update TABLE set STATUS='restarted' where idField=:captureId</pre>

Table 26: Database Capture Point Properties (Continued)

Property	Description
updatedUpdateSql	<p>The database update statement that updates specific attributes of a task in a special table in the source database when a task has been updated in iWD. The captureId parameter can be used to reference the particular row.</p> <p>For example:</p> <pre>update UPDATE_TABLE set PRIORITY=:priority where idField=:captureId</pre>
sourceUpdateQuerySql	<p>The database query that fetches a set of rows, where each row represents an update request coming from the source system. Each such update request may contain one or more columns that represent task attributes. The name of the column represents the name of the task attribute, but the value is the new value of that attribute.</p> <p>For example:</p> <pre>select id, channel, category, by businessValue from UPDATE_TABLE where processed='F'; id channel category businessValue ----- 1 fax new 200 2 e-mail new 150</pre>
sourceUpdatedUpdateSql	<p>The database update (or delete) query that will execute against a special table in the source database to mark a particular update as having been processed.</p> <p>For example:</p> <pre>update UPDATE_TABLE set processed='T' where idField=:captureId</pre>
sourceErrorUpdateSql	<p>This update is executed when there is an error executing an update request (the one that is fetched by sourceUpdateQuerySql). For example, if the task can not be found, then sourceErrorUpdateSql is executed and the error parameter is set to Cannot update task with captureId=<taskId>: task not found.</p>
captureBatchSize	<p>Performance tuning:</p> <p>The maximum number of rows that are to be returned by the query specified in the captureQuerySql parameter. A value of 0 sets the JDBC driver default value.</p>
sourceUpdateBatchSize	<p>Performance tuning:</p> <p>The maximum number of rows that are to be returned by the query specified in the sourceUpdateQuerySql parameter. A value of 0 sets the JDBC driver default value.</p>

Table 26: Database Capture Point Properties (Continued)

Property	Description
backupFor	High Availability: The primary Database Capture Point in a high availability setup scenario. This service will perform a backup role in such a setup, and will take over processing if the primary service becomes unavailable.
validateConnection	Enables and disables the validation of the connection to the capture database.
validationQuery	Enables a query to determine if the connection to the capture database is active. The default query works with Oracle databases. For other database servers, Genesys recommends that you use a query that returns at least one row. This is important because if your query returns zero rows the validation will fail. If you do have a query that returns zero rows, Genesys recommends that you complete the following steps: <ol style="list-style-type: none"> 1. Create an additional TABLE in your database that contains one integer field and one row. 2. Run the following statement: select * from <THAT_TABLE>. Note that the validationQuery property will only be visible once you select (check) the validateConnection property.

Rules Service

The Rules Service is a mandatory iWD service that allows iWD services to execute various rules.

In addition to the options and actions described in “Service Details” on [page 140](#), the properties listed in [Table 27](#) are configurable for the Rules Service.

Table 27: Rules 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 otherwise instructed by Genesys Technical Support. See “Service Log Levels” on page 152 for a description of each log level. Depending on the setting of this property, additional logging properties might be available. Refer to “Logging Service” on page 152 for descriptions of these common properties.

Business Context Management Service

The Business Context Management service is a mandatory service that works in conjunction with the Rules service to enable business rules to be invoked from within routing strategies:

- The Business Context Management service invokes the Rules service to execute rules for a particular interaction based on the ESP (Business Context Management Service) method that has been requested
- The Rules service executes rules, updates interaction attached data and passes this information back to Business Context Management Service and Interaction Server .

The Business Context Management Application is created by the iWD Setup Utility. The Business Context Management *service* needs to be created manually in iWD Manager.

Note: Refer to [page 221](#) for information about configuring multiple Business Context Management Services.

In addition to the options and actions described in “Service Details” on [page 140](#), the properties listed in [Table 28](#) are configurable for this service.

Table 28: Business Context Management Service Properties

Property	Description
startAutomatically	Whether the service should be started automatically after the configuration deployment.
logLevel	The Service log level. This should be set to Default unless otherwise instructed by Genesys Technical Support. See “Service Log Levels” on page 152 for a description of each log level. Depending on the setting of this property, additional logging properties might be available. Refer to “Logging Service” on page 152 for descriptions of these common properties.
CMEApplicationName	The name of the corresponding application in the Configuration database. This application was created by using the iWD Setup Utility. Refer to Procedure: Using the iWD Setup Utility , on page 69 .
configurationServerConnector	Mandatory dependency: The Configuration Server Connector service (see page 168); provides access to the Configuration Server.
RulesService	Mandatory dependency: The Rules service (see page 167) that is used to execute business rules.

Table 28: Business Context Management Service Properties (Continued)

Property	Description
workerThreads	Performance tuning: The size of the thread pool.
maxPriority	The maximum priority that can be assigned to a task. The priority will be automatically limited to this value if it exceeds it.

Database Service

The Database Service provides a connection to a database server. A typical iWD solution has one defined database connection for historical reporting data (iWD Data Mart).

The configured database and user must exist in the database server. The user must have read/write permissions to the database.

In addition to the options and actions described in “Service Details” on [page 140](#), the properties listed in [Table 29](#) are configurable for the Database Service.

Table 29: Database 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 <code>Default</code> unless otherwise instructed by Genesys Technical Support. See “Service Log Levels” on page 152 for a description of each log level. Depending on the setting of this property, additional logging properties might be available. Refer to “Logging Service” on page 152 for descriptions of these common properties.
DatabaseType	The type of the database server. Refer to the <i>Genesys Supported Operating Environment Reference Manual</i> for supported databases.
ServerName	The host name of the database server.
MysqlServerPort	The TCP port of the MySQL database server. This is available only when the DatabaseType is <code>MySQL</code> .
SQLServerCustomURL	If selected, the connection is configured by specifying a custom MS SQL JDBC URL. This option is available only when the DatabaseType is <code>MS SQL Server</code> .
SqlServerJDBCURL	The JDBC connection URL for MS SQL. This option is available when <code>SqlServerCustomURL</code> is selected.

Table 29: Database Service Properties (Continued)

Property	Description
SqlServerPort	The TCP port of the MS SQL database server. This option is available only when the DatabaseType is MS SQL Server.
DatabaseName	The name of the database. This is available only when the DatabaseType is MS SQL Server or MySQL.
OracleCustomURL	If this is checked, the connection is configured by specifying a custom Oracle JDBC URL. This option is available only when the DatabaseType is Oracle.
OracleJDBCURL	The JDBC connection URL for Oracle. This option is available when OracleCustomURL is selected.
OracleServerPort	The TCP port of the Oracle database server. This option is available only when the DatabaseType is Oracle.
OracleSID	Oracle System ID of the database. The Oracle System ID (SID) is used to uniquely identify a particular database on a system.
Username	The database user name.
Password	The password for the database.
AutoSynchronize	If selected, iWD will try to initialize/upgrade database structure automatically.

Statistics Adapter

The iWD Extended Statistics Service that existed in 7.6.1 has been merged into the Statistics Adapter service.

The Statistics Adapter Service processes the statistical data created by the Aggregate Stats ETL job and writes stat-types and filters in the configuration for Genesys Stat Server. CCPulse+ requests iWD statistics from Stat Server, and reads the stat-types and filters from the Stat Server configuration.

In addition to the options and actions described in “Service Details” on [page 140](#), the properties listed in [Table 30](#) are configurable for the Statistics Adapter Service.

Table 30: Statistics Adapter 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 otherwise instructed by Genesys Technical Support. See “Service Log Levels” on page 152 for a description of each log level. Depending on the setting of this property, additional logging properties might be available. Refer to “Logging Service” on page 152 for descriptions of these common properties.
triggerMode	<p>How the ETL job will be triggered (started):</p> <ul style="list-style-type: none"> • Trigger using CRON expression: Starts a job automatically, based on a CRON scheduling expression. • Trigger after another scheduled service: Starts a job automatically after another scheduled job has finished. • Trigger manually: Job can be started manually via the iWD Manager Services Status screen (see Figure 52 on page 139). <p>triggerMode and startAutomatically</p> <p>The startAutomatically property controls whether the service will be started at all. If it is unchecked, the service will not start on any of these trigger events.</p> <p>You can still start the service manually from the Service Status screen. In that case, the particular service will be in a wait state. It will wait for the trigger event to happen in order to start.</p> <p>Notes: If the triggerMode is set to Trigger manually and startAutomatically is unchecked, after deployment when you start the service it will be started, but it will wait for a manual trigger, so you will need to start it again so it can perform its tasks.</p> <p>If startAutomatically is checked and the triggerMode is set to Trigger using CRON expression, the service will start according to the CRON expression.</p>

Table 30: Statistics Adapter Service Properties (Continued)

Property	Description
cronExpression	<p>Standard CRON scheduling expression when triggerMode is set to Trigger using CRON expression. For example, the following expression will cause the job to be executed every 15 minutes:</p> <p><code>0 0, 15, 30, 45 * * * ?</code></p> <p>See http://www.quartz-scheduler.org/docs/tutorials/crontrigger.html for more information.</p>
executionQueueName	<p>The execution queue is configured only for jobs that are triggered with a cron expression or triggered manually.</p> <p>For chained jobs (Trigger after another scheduled service), the execution queue automatically assumes the value of the parent job. For example, if you have Load Intraday chained to Load Config and Aggregate Intraday chained to Load Intraday, Load Config will never start while Aggregate Intraday is still running.</p> <p>Scheduled services that have the same executionQueueName will never be run in parallel within the same solution.</p> <p>If a scheduled service is triggered while another scheduled service that has the same executionQueueName is in progress, it will be queued until the other service has completed its processing.</p> <p>Note: This only works within the boundaries of a single runtime node. If you have services on different nodes, this will have no effect across the nodes.</p>
dimensionMapping	<p>Defines how statistical dimensions are mapped.</p> <p>Filter: Dimensions are mapped to CCPulse+ filters.</p> <p>Virtual Queue: Dimensions are mapped to Genesys virtual queues.</p>
datamartDatabase	A reference to the Database service (see page 169) that points to the iWD Data Mart database.
virtualQueueName	Name of the Genesys virtual queue to which statistics are distributed.
configurationServerConnector	A reference to the Configuration Server Connector (see page 153); provides connectivity to Configuration Server.
CMETenantName	The name of the (Genesys Configuration Server) tenant in which the Stat Server application is defined.
statServerName	The name of the Stat Server application.

Table 30: Statistics Adapter Service Properties (Continued)

Property	Description
serviceIndex	Statistical service index for configuration options. This should be unique inside the set of indexes, assigned to statistical services served by the one instance of Genesys Stat Server.
extensionSectionName	Required to support a Genesys reporting environment with multiple instances of Stat Server Java Extensions. This property maps to the section name for the specific Stat Server Java Extension in Stat Server configuration.
extensionFileName	Required to support a Genesys reporting environment with multiple instances of Stat Server Java Extensions. This is the name of the Stat Server Java extension jar file (BPR_iWD_Extension.jar). This file was saved to the Stat Server installation directory during installation of the iWD Stat Extensions. You can find the location of this file in Stat Server configuration options as the value of the <code>java-libraries-dir</code> option in the <code>java-config</code> section.

Data Mart ETL Services

The Data Mart ETL services enable batch loading and aggregation of data into the iWD Data Mart. The Kettle ETL service defines general ETL configuration, such as the database in which the aggregated data will be stored. The Scheduled ETL Job service defines execution characteristics (such as scheduling) for each specific ETL job.

Kettle ETL Service

The Kettle ETL service represents an embedded Kettle ETL Engine that runs ETL scripts for loading and aggregating task and configuration data into the iWD Data Mart.

In addition to the options and actions described in “Service Details” on [page 140](#), the properties listed in [Table 31](#) are configurable for the Kettle ETL service.

Table 31: Kettle ETL 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 otherwise instructed by Genesys Technical Support. See “Service Log Levels” on page 152 for a description of each log level. Depending on the setting of this property, additional logging properties might be available. Refer to “Logging Service” on page 152 for descriptions of these common properties.
repositoryDirectory	The directory on the server in which iWD Data Mart ETL scripts are stored. For example, the default path used when iWD Data Mart is installed is C:\Program Files\GCTI\iWD Data Mart\etl.
customTaskAttributeMapping	<p>Up to 10 comma-separated names of a task's custom attributes that will be loaded into task_fact custom attribute fields (CUSTOM_ATTRIBUTE1-10). Names must start with a letters, and only underscores and alphanumeric characters are supported.</p> <p>In order for Kettle to pick them up, it is necessary to create fields in the Event Log Database. In the rpt_interaction and rpt_esp tables, add the fields in the following format:</p> <p>Name: <attribute_name>, type: varchar(length).</p> <p>These fields should be added to the mappings in the Event Log DAP options in the esp-custom-data and itx-custom-data sections.</p> <p>For example, in order to store a custom attribute with the name order_total in the iWD Data Mart, as a task custom attribute:</p> <ol style="list-style-type: none"> 1. Create a new column in the rpt_interaction table: order_total, type: varchar(50) 2. Create a new column in the rpt_esp table: order_total, type: varchar(50) 3. Create a new option in the esp-custom-data section of the Event Log DAP options: order_total=order_total 4. Create a new entry option in the itx-custom-data section of the Event Log DAP options: order_total=order_total 5. Add order_total to Kettle ETL service's customTaskAttributeMapping attribute list

Table 31: Kettle ETL Service Properties (Continued)

Property	Description
customTaskDimensionMapping	<p>Up to 5 comma-separated names of a task's custom attributes that will be loaded into the CUSTOM_DIM dimension and associated to the task via the CUSTOM_DIM_KEY field. Names must start with a letters, and only underscores and alphanumeric characters are supported.</p> <p>In order for Kettle to pick them up, it is necessary to create fields in the Event Log Database. In the rpt_interaction and rpt_esp tables, add the fields in the following format:</p> <p>Name: <attribute_name>, type: varchar(length).</p> <p>These fields should be added to the mappings in the Event Log DAP options in the esp-custom-data and itx-custom-data sections.</p> <p>For example, in order to store a custom attribute with the name order_total in the iWD Data Mart, as a task custom attribute:</p> <ol style="list-style-type: none"> 1. Create a new column in the rpt_interaction table: order_total, type: varchar(50) 2. Create a new column in the rpt_esp table: order_total, type: varchar(50) 3. Create a new option in the esp-custom-data section of the Event Log DAP options: order_total=order_total 4. Create a new entry option in the itx-custom-data section of the Event Log DAP options: order_total=order_total 5. Add order_total to Kettle ETL service's customTaskDimensionMapping attribute list
customTenantAttributeMapping	Up to 5 comma-separated names of a tenant's custom attributes that will be loaded into the CUSTOM_DIM dimension and associated to the tenant via the CUSTOM_DIM_KEY field.
customDepartmentAttributeMapping	Up to 5 comma-separated names of a department's custom attributes that will be loaded into the CUSTOM_DIM dimension and associated to the contract via the CUSTOM_DIM_KEY field.
customProcessAttributeMapping	Up to 5 comma-separated names of a process's custom attributes that will be loaded into the CUSTOM_DIM dimension and associated to the process via the CUSTOM_DIM_KEY field.
detailsExpirationDays	The number of days after which the detailed task (task_fact, task_event_fact, and task_work_fact) data will be removed from the database.
aggregation15minExpirationDays	The number of days after which the data will be removed from 15 minute aggregation tables.

Table 31: Kettle ETL Service Properties (Continued)

Property	Description
datamartDatabase	Mandatory dependency: Database (see page 169) in which to load and aggregate reportable data (Data Mart).
statisticsAdapter	Optional dependency: Statistics Adapter service (see page 170) to use for the delivery of statistics for current-intraday reporting.
timeZone	The time zone that will be used to store all date/time information in the fact tables.

Scheduled ETL Job

In addition to the options and actions described in “Service Details” on [page 140](#), the properties listed in [Table 32](#) are configurable for the Scheduled ETL Job service.

Table 32: Scheduled ETL Job Service Properties

Properties	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 otherwise instructed by Genesys Technical Support. See “Service Log Levels” on page 152 for a description of each log level. Depending on the setting of this property, additional logging properties might be available. Refer to “Logging Service” on page 152 for descriptions of these common properties.

Table 32: Scheduled ETL Job Service Properties (Continued)

Properties	Description
triggerMode	<p>How the ETL job will be triggered (started):</p> <ul style="list-style-type: none"> • Trigger using CRON expression: Starts a job automatically, based on a CRON scheduling expression. • Trigger after other scheduled service: Starts a job automatically after another scheduled job has finished. • Trigger manually: Job can be started manually via the iWD Manager Services Status screen (see Figure 52 on page 139). <p>triggerMode and startAutomatically</p> <p>The startAutomatically property controls whether the service will be started at all. If it is unchecked, the service will not start on any of these trigger events.</p> <p>You can still start the service manually from the Service Status screen. In that case, the particular service will be in a wait state. It will wait for the trigger event to happen in order to start.</p> <p>Notes: If the triggerMode is set to Trigger manually and startAutomatically is unchecked, after deployment when you start the service it will be started, but it will wait for a manual trigger, so you will need to start it again so it can perform its tasks.</p> <p>If startAutomatically is checked and the triggerMode is set to Trigger using CRON expression, the service will start according to the CRON expression.</p>
cronExpression	<p>Standard CRON scheduling expression when triggerMode is set to Trigger using CRON expression. For example, the following expression will cause the job to be executed every 15 minutes:</p> <p><code>0 0,15,30,45 * * * ?</code></p> <p>See http://www.quartz-scheduler.org/docs/tutorials/crontrigger.html for more information.</p>

Table 32: Scheduled ETL Job Service Properties (Continued)

Properties	Description
executionQueueName	<p>The execution queue is configured only for jobs that are triggered with a cron expression or triggered manually.</p> <p>For chained jobs (Trigger after another scheduled service), the execution queue automatically assumes the value of the parent job. For example, if you have Load Intraday chained to Load Config and Aggregate Intraday chained to Load Intraday, Load Config will never start while Aggregate Intraday is still running.</p> <p>Scheduled services that have the same executionQueueName will never be run in parallel within the same solution.</p> <p>If a scheduled service is triggered while another scheduled service that has the same executionQueueName is in progress, it will be queued until the other service has completed its processing.</p> <p>Note: This only works within the boundaries of a single runtime node. If you have services on different nodes, this will have no effect across the nodes.</p>
etlService	Mandatory dependency: The Kettle ETL Service (see page 173)
jobName	The name (type) of the ETL job. See Table 18, “Preconfigured ETL Jobs,” on page 148 for descriptions of the various types of iWD Data Mart ETL jobs. Additional information can be found in the <i>iWD 8.0 Data Mart Reference Guide</i> .

Configuration of iWD Current-Day Statistics in CCPulse+

This section explains how to configure iWD current-day statistics in CCPulse+. This section contains the following procedures:

- [Procedure: Configuring CCPulse+ for iWD](#)
- [Procedure: Creating a CCPulse+ template](#), on [page 179](#)

Procedure: Configuring CCPulse+ for iWD

Purpose: In Genesys, iWD current-day statistic measure types are presented as statistic objects, and iWD statistic dimensions are defined as filters. Therefore, it is necessary to combine statistic objects with filters in order to get a measure type for a dimension.

Prerequisites

- The iWD Data Mart database has been created, and the corresponding Database service has been configured and is running.
- The Aggregate Statistics ETL job is configured and running.
- Stat Server must be running, with the iWD Stat Extensions installed.
- The Statistics Adapter service must be configured and running.

Start of procedure

1. Start CCPulse+, and connect to the Genesys Stat Server (File > New).
2. In the Object Types dialog box, select the Virtual Queue object type for the corresponding Genesys tenant and switch, and click Next.
3. In the Object Instances dialog box, select the virtual queue that is used by the iWD to submit statistics, and click Finish.

Note: This is the virtual queue that was specified in [Procedure: Using the iWD Setup Utility](#), on [page 69](#).

End of procedure

Next Steps

- To be able to view iWD current-day statistics in Genesys CCPulse+, it is necessary to create a CCPulse+ template. The following procedure explains how to create a CCPulse+ template.

Procedure: Creating a CCPulse+ template

Purpose: To create a CCPulse+ template to use to view iWD current-day statistics.

Start of procedure

1. Open the CCPulse+ template wizard by selecting Template Wizard... from the Tools menu..
2. In the Template Definition dialog box, select the Virtual Queue object type in the Available Object Types list, then select the Create New Template option, and click Next.
3. In the Pre-defined Statistics dialog box, enter the template name, then click the New Group button, and enter a name for the newly created statistic group.

4. Expand the `TotalCustomValue` stat type in `Available Statistics` tree view, and select the iWD statistic that you want to add to the template (all iWD statistic names have a “GTL” prefix).
5. Click the `>>` button to add the selected statistic to the newly created statistic group.
6. In the `Requested Statistics` tree view, select the newly added statistic, and then click the `Properties` button.
7. In the `Statistic Properties` dialog box enter `Alias` (which is how the statistic is displayed in `CCPulse+`). Set `Insensitivity Value` to 1; set `Interval Type` to `Selection` and `GTLAggregated`; set `Notification Mode` to `Changes Based`; and set `Filter` to the filter type that represents the required iWD dimension; and then click `OK`.
8. Add more statistics and statistic groups, if necessary, and then click the `Next` button.
9. In the `Graph` dialog box, adjust graph parameters, if necessary, and then click `Finish`.
10. In the `CCPulse+` main window, select the virtual queue instance that is used by the iWD to submit statistics. Create a real-time view for this virtual queue:
In the `Real-Time Data Template` dialog box, select the previously created template from the list, and click `OK`.

End of procedure

Workforce Management Connector Service

The Workforce Management Connector Service allows you to use objects that exist in the WFM database, such as:

- Business Units and Sites that belong to Business Units
- Multi-Site Activities configured under Business Units
- Activities configured under Sites.

In order to use this service, an `Application` must be configured in `Configuration Server`.

Procedure: Configuring iWD to work with WFM

Purpose: To configure the Workforce Management Connector Service in iWD Manager, and add a connection to the WFM Server.

Prerequisites

- Workforce Management is installed and configured, as described in the *Workforce Management 8.0 Administrator's Guide*.
- The `attributeFilterExclude` property of the Interaction Server Connector service should contain the WFM Activity parameter (see Table 22 on [page 155](#) for a list of the Interaction Server Connector service's properties).

Start of procedure

1. Login to Configuration Manager or Genesys Administrator.
2. Locate the iWD Manager Application object and open its properties.
3. On the Connections tab, add a connection to the WFM Server application and save the change.
4. Login to iWD Manager.
5. Configure the corresponding Service in iWD Manager. Refer to “Service Details” on [page 140](#) for the common service options. Table 33 lists the additional configurable properties for the Workforce Management Connector Service.
6. Once the service is configured, WFM objects can be used in iWD.
7. In Departments and Processes, select any Capture Point, Department or Process, and add the action Assign WFM Activity. This action has two drop-down boxes. The first one contains Business Units and Sites, and the second one contains Multi-Site Activities and Activities.
8. If WFM Server is running, a list of Business Units and Sites for which the current user has permissions, is shown.
9. Select any Business Unit or Site.
10. Select a Multi-Site Activity or Activity from the second drop-down box.
11. When a task is generated you can see BU (or Site) in the WFMContext attribute and Multi-Site Activity (or Activity) in the WFMActivity attribute in the task's properties.

End of procedure

Table 33: Workforce Management Connector 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 otherwise instructed by Genesys Technical Support. See “Service Log Levels” on page 152 for a description of each log level. Depending on the setting of this property, additional logging properties might be available. Refer to “Logging Service” on page 152 for descriptions of these common properties.
CMEApplicationName	The name of the WFM Server as configured in Configuration Manager or Genesys Administrator. This is the WFM Server that was added to the Connections tab of your iWD Manager application.
configurationServerConnector	Mandatory dependency: The Configuration Server Connector service (see page 168); provides access to the Configuration Server.

Warning! When any changes are made to the WFM Connector Service (including creation of the service), the Undeployed Changes message does not appear in the Header bar. The changes do have to be deployed in order to take affect, so be sure to deploy your changes.

Departments and Processes

The Departments and Process configuration section is used to configure iWD business logic.

- Use Department Details view (see [page 183](#)) to create new departments, and to view and modify general department attributes and contract-level Metrics (see [page 203](#)).
- Use Process Details view (see [page 185](#)) to create new processes, and to view and modify general process attributes and process-level Metrics (see [page 203](#)).
- Use Rules view (see [page 191](#)) to create, view, and modify iWD business rules.

- Use Global Rules (see [page 192](#)) to create, view, and modify global business rules.
- Use Business Calendars view (see [page 203](#)) to create, view, and modify iWD business calendars.
- Use Deployment view (see [page 205](#)) to deploy solution configurations.

Department Details

Department Details view displays, and lets you modify, general department attributes. Refer to “Departments and Processes” on [page 182](#) for more information about departments. [Figure 57](#) shows the Department Details view.

MCRSolution1 > Customer Service

General **Rules**

ID T4_CSID **Department Name** Customer Service **Start Date** 05/05/2010 **End Date** 12/07/2015

Description
ACME Customer Service Department and associated business processes.

Contact Name John Smith **Contact Email** john.smith@acme.com **Contact Phone** +1 212 555 6789

Metrics (historical reporting)

Template	Metric	Distribution point	Value	Description
KPI Metrics	SLA Target		80.00 %	Department SLA Goal across all business processes

Custom Attributes (historical reporting)

Name	Type	Value	Description
Cost Center ID	Text	FRA	Cost center for customer service

Save Save & Close Cancel Delete View tasks Copy

Figure 57: Department Details

[Table 34](#) lists the attributes and actions that are available in the Department Details view.

Table 34: Department Details Attributes and Actions

Attribute/Action	Description
ID	The ID of the department.
Department Name	The name of the department. This field is mandatory.
Start Date	The date on which the department becomes active.
End Date	The last day that the department is active. If left empty, the period end date is unconstrained (that is, the department will be active infinitely).
Description	A free-form description of the department.
Contact Name, Phone and E-Mail	The contact information for the department, for informational purposes.
Metrics	A set of user-defined metrics, for reporting purposes (described in “Metrics” on page 203).
Custom Attributes	A set of user-defined attributes, for informational and reporting purposes. Each attribute must have a name, type (either Text, Number, Currency, or Date), value, and optional description.
Save/Save & Close/Cancel/Delete	Standard iWD Manager functions as described in “Object Actions” on page 108 .
View Tasks	Opens a corresponding task view in iWD Manager.

Process Details

Process Details view displays, and lets you modify, general process attributes. Refer to “Departments and Processes” on [page 182](#) for more information about processes. [Figure 58](#) shows the Process Details view.

McSolution1 > Customer Service > Complaint

General **Rules**

ID **Process Name** **Start Date** **End Date**
 T4_CSID_P4 Complaint 05/05/2010 12/07/2015

Description

Contact Name **Contact Email** **Contact Phone**

Metrics (historical reporting)

Template	Metric	Distribution point	Value	Description
KPI Metrics	Average Work-Tim		360 Seconds	Avg work time for tasks
KPI Metrics	SLA Target		89.90 %	% of tasks that should be completed within t

Custom Attributes (historical reporting)

Name	Type	Value	Description
Oracle Financials	Text	ORA-6789	Oracle financial code for reporting

Save **Save & Close** **Cancel** **Delete** **View tasks** **Copy**

Figure 58: Process Details

[Table 35](#) lists the attributes and actions that are available in Process Details view.

Table 35: Process Attributes and Actions

Attribute/Action	Description
ID	The ID of the process.
Process Name	The name of the process. This field is mandatory.
Start Date	The date on which the process becomes active.

Table 35: Process Attributes and Actions (Continued)

Attribute/Action	Description
End Date	The last day that the process is active. If left empty, the period end date is unconstrained (that is, the process will be active infinitely).
Description	A free-form description of the process.
Contact Name, Phone and E-Mail	The contact information for the process, for informational purposes.
Metrics	A set of user-defined metrics, for reporting purposes (described in “Metrics” on page 203).
Custom Attributes	A set of user-defined attributes for informational and reporting purposes. Each attribute must have a name, a type (either Text, Number, Currency, or Date), a value and an optional description.
Save/Save & Close/Cancel/Delete	Standard iWD Manager functions as described in “Object Actions” on page 108 .
View Tasks	Opens a corresponding task view in iWD Manager.

Business Calendars

A business calendar is a set of rules that define working days and hours, as well as holidays, that are applicable for the business. Business calendars can be used in iWD rules to perform date and time calculations, taking into account the working schedule of the business. Business calendars are defined on a solution level and can be used in rules only for that particular solution.

Managing Business Calendars in iWD Manager

To create new (or to manage existing) business calendars, open Departments & Processes view in iWD Manager, and select the Business Calendars entry in

the tree on the left-hand side of iWD Manager. [Figure 59](#) shows an example of a solution's business calendars.

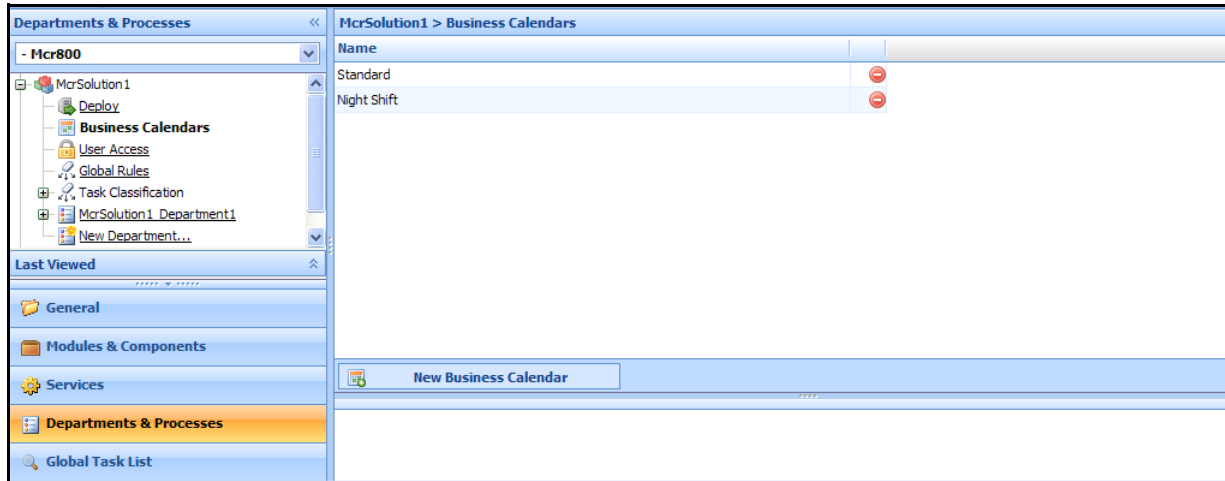


Figure 59: List of Solution's Business Calendars

The right-hand side of iWD Manager is split horizontally into two panes. The upper displays a list of a solution's business calendars. The New Business Calendar button that is underneath this list is used to create a new business calendar. You can delete business calendars by clicking the delete button that is on the right-hand side of the business calendar in the business calendar list. When a calendar in the business calendar list is selected, the lower pane displays the attributes of the selected business calendar. [Figure 60](#) shows an example of the attributes of a business calendar.



The screenshot shows the 'Standard' business calendar attribute form. The form has a title bar 'Standard' with a close button. Below the title bar are fields for 'ID' (BC0) and 'Name' (Standard). The 'Timezone' field is a dropdown menu showing 'America/Moncton (GMT-4) (+DST)'. Below the timezone field are four fields: 'Week starts on' (Monday), 'Week ends on' (Friday), 'Start time' (09:00), and 'End time' (17:00). Below these fields is a section titled 'Business calendar rules' with a table. The table has four columns: 'Name', 'Entry type', 'Calendar placement', and 'Definition'. The table is currently empty. At the bottom of the form are four buttons: 'Save', 'Save & Close', 'Cancel', and 'Copy'.

Figure 60: Attributes of a Business Calendar

Business calendars consist of a set of standard mandatory attributes, as well as optional business calendar rules. [Table 36](#) describes the standard attributes of a business calendar.

Table 36: Business Calendar Attributes

Attribute name	Description
ID	The business calendar identifier. This attribute is generated automatically by iWD Manager when the calendar is created.
Name	The name of the business calendar as it will appear throughout iWD Manager.
Timezone	The time zone of the business calendar.
Week starts on	The first day of a working week.
Week ends on	The last day of a working week.
Start time	The start time of a working day. This attribute is a representation of time in 24-hour format (<i>hh:mm</i> , where <i>hh</i> are hours and <i>mm</i> are minutes).
End time	The end time of a working day. This attribute is a representation of time in 24-hour format (<i>hh:mm</i> , where <i>hh</i> are hours and <i>mm</i> are minutes).

Business calendar rules describe exceptions to the regular working schedule that is defined by business calendar standard attributes. It is possible to add and delete business calendar rules by clicking the  and  icons located on the right side of the business calendar rule list. Depending on calendar placement

and entry type, the definition of a business calendar rule can have different attributes. [Table 37](#) describes the attributes of a business calendar rule.

Table 37: Business Calendar Rule Attributes

Attribute	Description
Name	The name of the business calendar rule.
Entry type	<p>The type of business calendar rule. This can be one of the following:</p> <p>Holiday: The business calendar rule describes a holiday.</p> <p>Time Change: The business calendar rule describes working hours that differ from the working hours of the business calendar.</p> <p>Note: A business calendar's holidays precede time changes; therefore, if any rule sets a day as a holiday, and there is a time change for that specific day, the time change will have no effect. This applies only to holidays that are specified by the rules of the business calendar.</p>
Calendar placement	<p>The calendar placement of the business calendar rule. This can be one of the following:</p> <p>Fixed: The business calendar rule is tied to a specific date.</p> <p>Relative: The business calendar rule is described in a relative way and is not tied to a specific date; for example, the first Monday of September would be a relative description.</p> <p>Annual: The business calendar rule that happens every year on a specific date.</p>

Table 37: Business Calendar Rule Attributes (Continued)

Attribute	Description
Definition	<p>The contents of this attribute depend on the selected calendar placement and entry type:</p> <p>Date input control: This parameter applies only to business calendar rules that have a fixed calendar placement. The year portion of this date is not taken into account, when performing calculations; thus, making this date recurrent on yearly basis.</p> <p>Relative calendar placement controls: This set of controls applies only to business calendar rules that have a relative calendar placement. There are three drop down list controls that allow specification of Place in month, Day of week, and Month of year. Place in month is defined by a value of Every, First, Second, Third, or Fourth, and is used in conjunction with Day of week (Monday, Tuesday, and so on) to describe relatively the day of month. Month of year defines to what month a business calendar applies; besides month names, it allows also specification of “any month,” if the business calendar rule must be recurrent on a monthly basis. The following are examples of definitions of a relative calendar placement:</p> <p>[Second] [Tuesday] of [September] [Every][Saturday] of [May] [First][Monday] of [any month]</p> <p>Annual calendar placement controls: This set of controls applies only to business-calendar rules that have an annual calendar placement. There are two dropdown list controls that allow specification of month and date.</p> <p>Working hours definition controls: This set of controls applies only to business calendar rules that have a time-change type. Two input controls allow the entering of a start time and end time of the working day. Each value is a representation of time in 24-hour format (<i>hh:mm</i>, where <i>hh</i> are hours and <i>mm</i> are minutes).</p>

Simple example

The following procedure explains how to configure a simple business calendar.

Procedure: Configuring a business calendar

Purpose: To configure a business calendar in which Monday-Friday are standard working days 09:00 to 18:00, Saturday 10:00 to 16:00.

Start of procedure

1. Create a new calendar, select Timezone, and set Week starts on Monday and Week ends on Friday. Then, set Start time and End time.
2. Add a time-change rule for Saturdays (it will mark it as working day).
3. Some holidays can be specified with rules, such as Christmas or New Year's Day. See [Figure 61](#).

End of procedure

Name	Entry type	Calendar placement	Definition	
Saturdays	Time Change	Relative	Every Saturday of any month Start time: 10:00 End time: 16:00	⊕ ⊖
New Year's Day	Holiday	Annual	January 1	⊕ ⊖
Christmas Day	Holiday	Annual	December 25	⊕ ⊖

Figure 61: Example Business Calendar

Rules

In iWD Manager, when a department, process, capture point, or global rule is selected in the navigation tree, the rules for that particular context are shown on the right side of iWD Manager, under the Rules tab. Rules are presented in a list, in the same order as they are applied to tasks. You can change the order of rules by clicking the Up and Down buttons. The logic of a particular rule can be expressed as either a linear rule or a decision table.

Any interaction attribute can be read or updated by business rule conditions and actions, respectively.

Global Rules iWD 8.0 supports “Global Rules”. Global business rules allow you to specify classification and prioritization rules for the entire iWD Solution. They provide:

- the ability to configure classification rules globally instead of per Capture Point.
- the ability to configure prioritization rules globally instead of per Department or Process.
- support for tasks (interactions) not captured by iWD Capture Points. This includes Genesys as well as third-party media servers, such as Genesys E-mail Server, G+ Adapters, and custom integrations previously built using Open Media API.

Global rules are applied before any other rules. This means that for classification, the sequence is:

1. Global rules
2. Capture point rules
3. Department rules
4. Process rules

Note: Any of these steps are optional. However after Step 3 a process has to be assigned, otherwise the task status will be changed to `ERROR_HELD`. For example, a department can be assigned in a global rule or capture point rule, followed by assigning a process at the department level. Alternatively, a process may be directly assigned in a global rule or capture point rule. If a department or process is assigned in both global and capture point rules based on the same conditions, the capture point rules will override.

For prioritization the sequence is:

1. Global rules
2. Department rules
3. Process rules

Linear Rule

A linear rule is a business rule that has a set of conditions (when) and actions (then), and is used for a simple (linear) business case. For example, when a

task is due in 1 to 8 hours, set the task's priority to 20. [Figure 62](#) shows an example.

The screenshot shows the 'Rules' tab in the iWD Manager interface. The breadcrumb path is 'McrSolution1 > Customer Service > Complaint'. The 'Rules' tab is active, showing a table with one rule entry:

ID	Name	Phase	Start Date	End Date
LR_5	Set iWD Priority to 20 if task is due in 1 to 8 hours	classification	05/05/2010	05/05/2010

Below the table are buttons for 'New Decision Table...' and 'New Linear Rule...'. The 'New Linear Rule...' button is highlighted. Below these buttons is a section titled 'Set iWD Priority to 20 if task is due in 1 to 8 hours' with a sub-header 'Expression' and 'Parameters'. At the bottom of the window are buttons for 'Add condition', 'Add action', 'Save', 'Save & Close', 'Cancel', and 'Copy'.

Figure 62: Linear Rule

The following procedure explains how to specify a linear rule.

Procedure: Specifying a linear rule

Purpose: To specify a linear rule for a simple business case.

Start of procedure

1. In iWD Manager, in the Department & Processes navigation tree, select a department, process, capture point, or global rule.
2. On the right side of iWD Manager, in the Rules tab, click New Linear Rule.
3. In the rule Name, enter a name that identifies the rule.
4. Select the Phase in which to apply the rule. Depending on the current context (capture point, department, process, or global rule) this can be either classification, prioritization, or archiving.
5. If needed, set the Start Date and End Date.
Leaving these empty means that the rule activation period is unconstrained.

6. Select one or several conditions for the rule from the Add condition combo box.
Conditions are logically concatenated by using the logical AND function.

Note: Available actions and conditions are defined in Rules templates. If a new action or condition type is required, add it to either an existing rules template or a new template.

7. Select one or several actions for the rule from the Add action combo box.
8. To save the specified rule, click either Save or Save & Close.

End of procedure

Decision Table

Decision tables have a set of the same conditions (when) and actions (then), but have different parameters and are used for a complex (structured) business case.

Use decision tables to avoid dozens of linear rules in the system. Defining a decision table is similar to defining a linear rule. The following procedure explains how to define a decision table.

Procedure: Defining a decision table

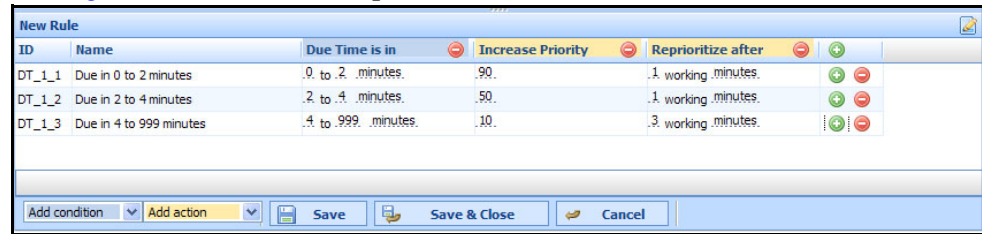
Purpose: To define a decision table for a complex business case.







Start of procedure

1. In the Rules tab, click New Decision Table.
2. Specify the rule Name, Phase, Start Date, and End Date, as required (in the same way as for a linear rule).
3. Click the add rule icon, and then enter the name of the rule.
4. Select one or several conditions for the rule from the Add condition combo box. For each selected condition, set the parameter value.
5. Select one or several actions for the rule from the Add action combo box. For each selected action, set the parameter value.
6. To delete condition, action, or rule, click the Delete icon, which is located on the right side of the column (for conditions and actions), or on the right side of the row (for rules).

7. Repeat [Steps 3](#) through [5](#), until you have set all of the required cases. The result is a table in which the columns represent rule conditions and actions and the rows contain real conditions and action parameter values.

[Figure 63](#) shows an example.



ID	Name	Due Time is in	Increase Priority	Reprioritize after	
DT_1_1	Due in 0 to 2 minutes	.0 to .2 .minutes.	.90.	.1 working .minutes.	 
DT_1_2	Due in 2 to 4 minutes	.2 to .4 .minutes.	.50.	.1 working .minutes.	 
DT_1_3	Due in 4 to 999 minutes	.4 to .999 .minutes.	.10.	.3 working .minutes.	 

At the bottom of the dialog, there are buttons for 'Add condition', 'Add action', 'Save', 'Save & Close', and 'Cancel'.

Figure 63: Decision Table

8. To save the specified rule, click either Save or Save & Close.

End of procedure

Standard Rules Template

The Standard Rules Template defines the most commonly used rule conditions, actions, parameters, and functions..

Note: The ACME sample (see [page 94](#)) contains sample business rules that show how some of the Standard Rules Template conditions and actions can be used in practice. This includes assigning a task to an iWD process based on specific attributes of the task such as Product Type, as well as setting priority and business value based on the iWD process. The sample also includes examples of how to set the reprioritization interval in a rule.

Conditions

Table 38 describes conditions for the Standard Rules Template.

Table 38: Standard Rules Template Conditions

Condition	Explanation
Business value is “ <i>from</i> ” to “ <i>to</i> ”	If the business value of the task is between <i>from</i> and <i>to</i> , then... Note: This rule condition is designed to test inclusive conditions. That is, the “ <i>from</i> ” and “ <i>to</i> ” conditions that are being checked are “>=” and “<=”, respectively. This must be understood in order to use this rule condition effectively.
Due Time is in “ <i>from</i> ” to “ <i>to</i> ” “ <i>units</i> ”	If the task due date/time is between <i>from</i> and <i>to</i> specified time <i>units</i> , then... Note: This rule condition is designed to test conditions that are not inclusive. That is, the “ <i>from</i> ” and “ <i>to</i> ” conditions that are being checked are “>” and “<”, respectively. This must be understood in order to use this rule condition effectively.
Priority is “ <i>operator</i> ” “ <i>priority</i> ”	If iWD priority of the task is equal to/not equal to/less than/less than or equal to/greater than/greater than or equal to <i>value</i> , then...
Channel is “ <i>channel</i> ”	If the specified channel of the task is <i>channel</i> , then...
Department is “ <i>department</i> ”	If the specified department of the task is <i>department</i> , then...
Integer “ <i>attribute</i> ” “ <i>operator</i> ” “ <i>integerValue</i> ”	When the value of a specified custom attribute of type integer is greater than/less than/equal to the specified <i>integerValue</i> , then...
Is first prioritization	When the rules being applied are part of the initial prioritization step (the value of the task attribute <code>reprioritizeDateTime</code> is empty), then...
Is reprioritization	When the rules being applied are part of the reprioritization step (the value of the task attribute <code>reprioritizeDateTime</code> is not empty), then...
Media type is “ <i>mediaType</i> ”	When the specified media type of the task is <i>mediaType</i> , then...

Table 38: Standard Rules Template Conditions (Continued)

Condition	Explanation
String “ <i>attribute</i> ” equals “ <i>stringValue</i> ”	When the value of a specified custom attribute of type <code>string</code> is equal to the specified <i>stringValue</i> , then...
No process selected	If process is not assigned to the task, then..
Process is “ <i>process</i> ”	If the task is assigned to <i>process</i> , then...
Task is overdue	When the task is overdue, then...

Actions

[Table 39](#) describes Standard Rules Template actions.

Table 39: Standard Rules Template Actions

Action	Explanation
Activate task in “ <i>amount</i> ” “ <i>units</i> ”	Activate the task in the specified <i>amount</i> of time <i>units</i> .
Activate task in “ <i>amount</i> ” working “ <i>units</i> ”	Activate the task in the specified <i>amount</i> of time <i>units</i> according to the task's business calendar (working days, working hours and so on). Note: Business calendar must be assigned to the task before this action can be used.
Archive destination “ <i>archive</i> ”	Set the archive destination for the task.
Assign business calendar “ <i>business calendar</i> ”	Assign the specified <i>business calendar</i> to the task.
Assign distribution point “ <i>distribution point</i> ”	Assign the specified <i>distribution point</i> to the task.
Assign iWD department “ <i>department</i> ”	Assign the task to the specified <i>department</i> .
Assign iWD process “ <i>process</i> ”	Assign the task to the specified <i>process</i> .

Table 39: Standard Rules Template Actions (Continued)

Action	Explanation
Assign WFM Activity “wfmActivity”	Assign the task to the specified Genesys <i>Workforce Management Activity</i> . Note: For iWD to read objects from Genesys Workforce Management, the Workforce Management Connector Service (see page 180) must be configured.
Increase iWD priority “amount”	Increase the iWD priority of the task by the specified <i>amount</i> .
Multiply business value “amount”	Multiply the business value of the task by the specified <i>amount</i> .
Reprioritize after “period” “periodType”	Reapply prioritization rules to the task after the specified <i>amount</i> of time <i>units</i> from the current time.
Reprioritize after “period” working “periodType”	Reapply prioritization rules to the task after the specified amount of time units, according to the task's business calendar (working days, working hours, and so on). Note: Business calendar must be assigned to the task before this action can be used.
Request agent “agent”	Request a specific <i>agent</i> for the task.
Request skill “skill”	Request a specific <i>skill</i> for the task.
Request agent group “agentGroup”	Specify which <i>agent group</i> is required to process the task.
Request place group “placeGroup”	Specify which <i>place group</i> is required to process the task.
Set activation date from “customAttribute”	Set the activation date/time of the task from the specified <i>custom attribute</i> of the task.
Set activation time “time”	Set the <i>time</i> when the task will be activated.
Set business value “businessValue”	Set business value of the task to the specified <i>value</i> .
Set due date from “customAttribute”	Set the due date/time of the task from the specified <i>custom attribute</i> of the task.

Table 39: Standard Rules Template Actions (Continued)

Action	Explanation
Set due time “ <i>time</i> ”	Set the <i>time</i> when the task is due.
Set integer “ <i>attribute</i> ” value “ <i>integerValue</i> ”	Set the value of a specified custom attribute of type integer to the specified <i>integerValue</i> .
Set priority “ <i>priority</i> ”	Set the iWD priority of the task to the specified <i>value</i> .
Set string “ <i>attribute</i> ” value “ <i>stringValue</i> ”	Set the value of a specified custom attribute of type string to the specified <i>stringValue</i> .
Task Due in “ <i>period</i> ” “ <i>periodType</i> ”	The task is due in, after the specified <i>amount</i> of time <i>units</i> .
Task Due in “ <i>period</i> ” working “ <i>periodType</i> ”	The task is due in, after the specified <i>amount</i> of time <i>units</i> , according to the task's business calendar (working days, working hours, and so on). Note: A business calendar must be assigned to the task before this action can be used.
Task expires in “ <i>period</i> ” “ <i>periodType</i> ”	Sets the task's expiration date/time after the specified <i>amount</i> of time <i>units</i> .
Task expires in “ <i>period</i> ” working “ <i>periodType</i> ”	Sets the task's expiration date/time after the specified <i>amount</i> of time <i>units</i> , according to the task's business calendar (working days, working hours, and so on). Note: A business calendar must be assigned to the task before you can use this action.

Parameters

Parameters in a Rules template describe the nature of the parameters that are used in rule conditions and actions. This includes how parameters are presented to the user (who is creating the rule), how input is validated, and so on.

Table 40 describes parameters for the Standard Rules Template.

Table 40: Rules Template Parameters

Parameter	Description
periodType	Presents you with predefined types of time periods, such as <i>days</i> , <i>hours</i> or <i>minutes</i> .
periodTypeBC	Presents you with predefined types of time periods used in business calendars, such as <i>working days</i> , <i>working hours</i> , or <i>working minutes</i> .
businessValue	Allows you to enter the numeric value that represents business value.
period	Allows you to enter the numeric value. Combined with the period type, it gives the actual value of the time period.
priority	Allows you to enter the numeric value that represents iWD priority.
process	Presents you with a list of processes that are defined in the solution. This list is dynamic; it changes as processes get added or removed.
periodFrom	Allows you to enter the numeric value that represents the start point of a period, in time units, according to period type.
periodTo	Allows you to enter the numeric value that represents the end point of a period, in time units, according to period type.
customAttribute	Allows you to enter text that represents the name of a task custom attribute.
skill	Presents you with a list of skills that are read from the Genesys configuration database. This list is dynamic; it changes as skills get added or removed.
businessCalendar	Presents you with a list of business calendars that are defined in the solution. This list is dynamic; it changes as business calendars get added or removed.
wfmActivity	The associated WFM activity.
mediaType	The media type.

Table 40: Rules Template Parameters (Continued)

Parameter	Description
operator	Provides operators <i>Equal To</i> , <i>Not Equal To</i> , <i>Less Than</i> , <i>Less Than or Equal To</i> , <i>Greater Than</i> , and <i>Greater Than or Equal To</i> , that can be used to compare values of custom attributes of type integer, to values specified in a rule condition.
department	Presents you with a list of departments that are defined in the solution. This list is dynamic; it changes as processes are added or removed.
taskChannels	Presents you with a list of predefined task channels, read from an iWD Lookup Table.
agent	Presents you with a list of agents' user names that are read from the Genesys configuration database. This list is dynamic; it changes as agents are added or removed.
agentGroup	Presents you with a list of Agent Groups that are read from the Genesys configuration database. This list is dynamic; it changes as agents are added or removed.
placeGroup	Presents you with a list of Place Groups that are read from the Genesys configuration database. This list is dynamic; it changes as agents are added or removed.
distributionPoint	Presents you with a list of distribution points that are defined in the solution. This list is dynamic; it changes as entries in <code>distributionPoint</code> lookup table are added or removed.
integerValue	Allows you to enter an integer value for use in rule conditions or actions that evaluate the value of tasks custom attributes.
stringValue	Allows you to enter a string value for use in rule conditions or actions that evaluate the value of tasks custom attributes.
attribute	Allows you to enter text that represents the name of a task attribute.
archive	Presents you with a list of possible archive destinations for expired tasks.

Functions

Table 41 describes functions for the Standard Rules Template.

Table 41: Rules Template Functions

Function	Description
getCurrentDT	Returns the current date and time, in UTC.
getPeriodDTFrom	Adds the specified amount of time units (minutes, hours, days) to the specified date/time value.
setDepartmentAndProcess	Sets the department and process of the given task from a given string, in <i>department/process</i> format.
isProcess	Is true if a given task is assigned to a given <i>process</i> , false otherwise.
setTime	Sets the time portion of a given date/time value from a string value, in <i>hh:mm</i> format.
getDueDT	Returns the task due date or throws an exception if a due date is not set.
compareInteger	Compares the value of two integers.
setIntValue	Sets the value of a task attribute as an integer.
getIntValue	Gets the value of a task attribute as an integer.
setStringValue	Sets the value of a task attribute as a string.
getStringValue	Gets the value of a task attribute as a string.
increaseIntegerValue	Increases the value of an integer by adding to it.
multiplyIntegerValue	Increases the value of an integer by multiplying it by some factor.
getDTValue	Gets the value of a task attribute as date/time.
setDTValue	Sets the value of a task attribute as date/time.
notNull	Is true if a task attribute value is Not Null, false otherwise.
isNull	Is true if a given task attribute value is Null, false otherwise.
getWFMActivityContext	Retrieves WFM Activity context (Business Unit or Site).
getWFMActivity	Retrieves WFM Activity.

Metrics

A key component of dashboards and reports is the comparison of actual metrics against target goals. Understanding the effectiveness or efficiency of organizations requires measuring performance against important goals that have been set by the organization.

Targets can be associated with a number of objects in iWD Manager, such as processes, departments, or tenants. For example, a work-time goal for a task will differ, based on its process; for example, orders will take longer than address changes. [Figure 64](#) shows an example.

Template	Metric	Distribution point	Value	Description
KPI Metrics	Average Work-Time		360 Seconds	Avg work time for tasks
KPI Metrics	SLA Target		89.90 %	% of tasks that should be completed within t

Figure 64: Metrics

Metrics can be configured at the business-user level for departments and processes using metric types that are defined in available Metrics templates (see “Metrics Templates” on [page 135](#)). When a metrics value is set, it will be stored as a named attribute in Data Mart. If the value is changed, the updates are pushed through to Data Mart with a `valid_from` and `valid_to` date/time stamp. This is important for historical reporting; example, if you update the target on November 1 from 2.5 to 3.5, all tasks up to November 1 will use 2.5, and all new tasks will use 3.5.

If the value is set at a department level, it applies to all processes, unless there is a specific value for that process. For example, Department 1 has four processes: A, B, C, and D. Cost/Task @ Department 1 = 2.50, which applies to Processes B, C, and D. Cost/Task @ Process A = 1.50, which applies only to Process A.

Using business calendars in iWD rules

After business calendars are defined, it is possible to use them in iWD rules. Business calendars should be assigned to a task before any business calendar related calculations can be performed on task values. [Table 42](#) describes the

business calendar-related actions that are available in the Standard Rules Template.

Table 42: iWD Rules Actions

Action	Parameters	Description
Assign business calendar	{businessCalendar}	Assigns a business calendar to a task. A business calendar must be assigned to a task, before any business calendar-related calculations can be performed on task values. A dropdown list displays a list of business calendars that are defined for the solution.
Reprioritize after	{period} working {periodType}	Sets a task's re-prioritization date/time to value that is calculated, based on current date/time, the task's business calendar, and specified parameters. {period} is a numeric value, and {periodType} specifies “working” minutes, hours, or days.
Task Due in	{period} working {periodType}	Sets a task's due date/time to a value that is calculated, based on the task's creation date/time, the task's business calendar, and specified parameters. {period} is a numeric value, and {periodType} specifies “working” minutes, hours, or days.

Table 42: iWD Rules Actions (Continued)

Action	Parameters	Description
Activate task in	{period} working {periodType}	Sets a task's activation date/time to a value that is calculated, based on task's creation date/time, the task's business calendar, and specified parameters. {period} is a numeric value, and {periodType} specifies "working" minutes, hours, or days.
Task expires in	{period} working {periodType}	Sets a task's expiration date/time to a value that is calculated, based on task's creation date/time, the task's business calendar, and specified parameters. {period} is a numeric value, and {periodType} specifies "working" minutes, hours, or days.

In addition to these standard rule actions that use business calendars, you may build other rule expressions that use business calendar functions. For more details, see <https://sites.google.com/a/iwdlab.com/iwd8/rules/bc>.

Deployment

Deployment in the Departments and Processes configuration section is identical to Deployment in the Services configuration section. Refer to "Deployment" on [page 143](#) for more information.

Solution User Access

Departments and processes support instance-level security. The User Access page under Solution allows the assignment of configured User roles to specific departments or processes. Any roles that have been specified on a department level are automatically inherited by all child processes. Process roles are appended to inherited ones.

User access can be copied to another solution, if there are departments or processes that have matching IDs.

Note: This user access only pertains to access to iWD business context elements, through iWD Manager. It does not relate directly to Genesys Configuration Access Groups or other Configuration Server object permissions.

5

iWD Manager and Task Management

This chapter describes how to monitor and manage tasks in intelligent Workload Distribution (iWD).

This chapter contains the following section:

- [Task Management, page 207](#)

Task Management

Task Management allows monitoring and management of tasks that are being processed by iWD:

- Use Task Monitoring (see [page 207](#)) to view a list of tasks that are associated with different business contexts, as well as details and history for each task.
- Use Task Operation (see [page 213](#)) to override configured task-handling logic by performing a manual task operations on specific tasks such as Hold, Resume, Cancel, and Modify.
- Use Filters (see [page 215](#)) to refine the list of tasks that are available in Task Monitoring by defining filter criteria and visible task attributes (columns).

Task Monitoring

Task Monitoring allows you to view a current list of tasks for a number of business contexts:

- Solution
- Department

- Process
- Capture Point

All of the available contexts are displayed in the navigation tree, as shown in [Figure 65](#). When a context is selected, the corresponding list of tasks is displayed in the Global Task List.

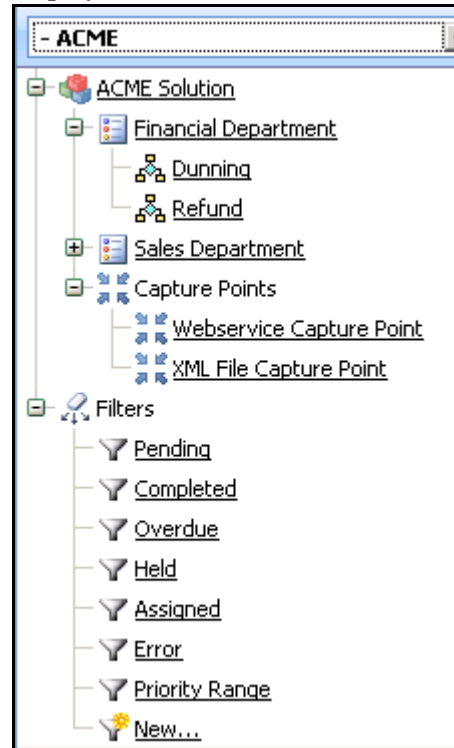


Figure 65: Navigation Tree

Global Task List

The Task List displays a list of tasks for the selected business context, as illustrated below in [Figure 66](#).

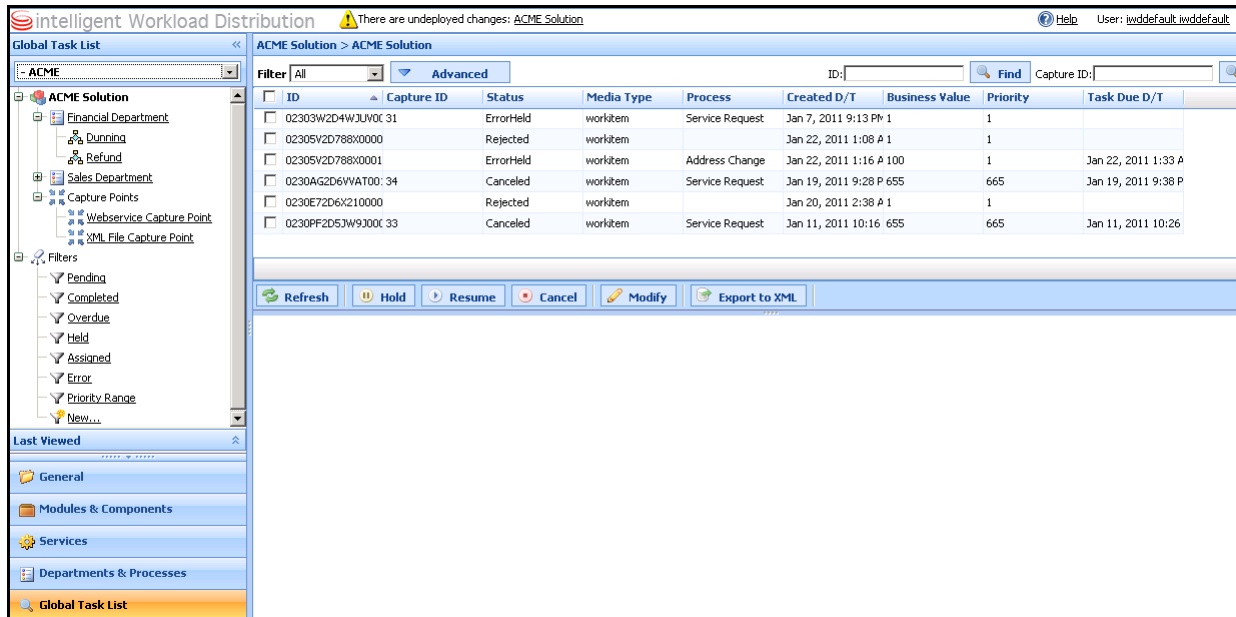


Figure 66: Global Task List

[Table 43](#) lists the actions that are available in the Global Task List.

Table 43: Global Task List Actions

Attribute/Action	Description
Filter	Refines the Global Task List and displayed columns, based on the selected filter. For more information on filters, see “Filters” on page 215 .
Advanced Filters	Advanced filters let you further narrow down the tasks that are listed in the Global Task List. Up to three additional conditions can be added to a selected filter. All of the task’s core and extended attributes are available. To add an advanced filter, select the attribute name from the drop down list. Advanced filters cannot be saved and are retained only during an iWD Manager session.
Refresh	Refreshes the Global Task List.

Table 43: Global Task List Actions (Continued)

Attribute/Action	Description
Hold, Resume, Cancel, Modify, Export to XML	Performs the corresponding operation on the selected task. The task is selected if the first column of the task row in the list is checked. To select/deselect all tasks in the list, click the first column in the table header. If no task is checked, but task details are opened for some task, this task is also considered to be selected. For more information on specific operations, see “Task Operations” on page 213 .
Configuration	Opens the configuration view of the current business context. This action can be restricted by a security policy.
Click task row	Displays task details. The selected row is marked as illustrated in Figure 66 .

Task Details

When a task is selected from the Global Task List, its attributes will be displayed. [Figure 67](#) shows an example of the Task Details.

ACME Solution > ACME Solution

Filter: All Advanced ID: Find Capture ID: Find

ID	Capture ID	Status	Media Type	Process	Created D/T	Business Value	Priority	Task Due D/T
<input type="checkbox"/> 02305V2D788X0001		ErrorHeld	workitem	Address Change	Jan 22, 2011 1:16 A 100		1	Jan 22, 2011 1:33 A
<input checked="" type="checkbox"/> 0230AG2D6VAT001: 34		Canceled	workitem	Service Request	Jan 19, 2011 9:28 P 655	665		Jan 19, 2011 9:38 P
<input type="checkbox"/> 0230E72D6X210000		Rejected	workitem		Jan 20, 2011 2:38 A 1		1	

Refresh Hold Resume Cancel Modify Export to XML

Task Details

Attributes History

ID: 0230AG2D6VAT001 Status: Canceled

Media Type: workitem Tenant: ACME

Department: Sales Department Process: Service Request

Category: Capture Point: Webservice Capture Point

Capture ID: 34 Created D/T: Jan 19, 2011 9:28 PM

Channel: webform Distribution Point:

Age: 18m 49s Activation D/T:

Business Value: 655 Task Due D/T: Jan 19, 2011 9:38 PM

Priority: 665 Reprioritization D/T: Feb 9, 2011 12:35 AM

Queue: IWD_Canceled Queue Type: InteractionQueue

Queue Target: Assigned To: invdefault

Assigned D/T: Feb 9, 2011 12:34 AM Completed D/T: Jan 19, 2011 9:47 PM

Expiration D/T: Feb 18, 2011 9:28 PM Business Calendar:

Customer Segment: Customer ID:

Product: Subproduct:

TOS Tenant: TOS Process: Service request

TOS Subprocess: Corporate Information TOS Created D/T:

Figure 67: Task Details

Attributes are broken down into three sections:

- Top—core task attributes
- Middle—extended task attributes (displayed only when the task has extended attributes)

- Bottom—custom task attributes. These attributes can be filtered by configuring the `attributeFilterInclude` and `attributeFilterExclude` properties of the Interaction Server Connector service (see [page 154](#)).

For attributes that display a timestamp such as Task Due D/T, it is possible to view the offset from the current time by moving the mouse cursor on top of the attribute. The offset is displayed in a hint.

Task History

Task History can be viewed by clicking the History tab in the Task Details view. It displays all of the events that are related to the task that has been selected in the Global Task List. [Table 44](#) lists the attributes and actions that are available in the Task History.

Table 44: Task History Attributes

Attribute/Action	Description
Date/Time	The date/time when the event occurred.
Actor	Shows who triggered the event. This is empty for iWD system-triggered events.

Table 44: Task History Attributes (Continued)

Attribute/Action	Description
Event Code	<p>Represents the type of the event:</p> <p>NEW: Task has just been created.</p> <p>CLASSIFY_START: Task classification has started.</p> <p>CLASSIFY: Task classification has finished.</p> <p>PRIORITIZE_START: Task initial prioritization has started.</p> <p>PRIORITIZE: Task initial prioritization has finished.</p> <p>HOLD: Task is held. This can be triggered by the task source system or by the user with the task Held operation (see page 213).</p> <p>RESUME: Task is resumed from the held state. This can be triggered by the task source system or by the user with the task Resume operation (see page 213).</p> <p>REJECT: Task was rejected. Either the contract or the process to which the task was associated is expired or not yet active.</p> <p>ERROR: A processing error has occurred, and the task was held for that reason.</p> <p>UPDATE: Task attributes are updated. This can be triggered by a task source system or based on updates to a task's attached data from a routing strategy or Genesys Interaction Server client such as an agent or knowledge worker's desktop application.</p> <p>UPDATE_COMPLETE: Task attributes are updated after a task is completed.</p> <p>ASSIGN: Task is assigned to an agent.</p> <p>FINISH: Agent has finished working on the task.</p> <p>FINISH_RETURN: Agent has returned the task to queue.</p> <p>STOPPED: That task has stopped (removed from database).</p> <p>COMPLETE: Task is completed.</p> <p>QUEUE: The task is queued.</p> <p>DISTRIBUTE_QUEUE: The task is put into a queue.</p> <p>DISTRIBUTE_WORKBIN: The task is put into to a workbin.</p> <p>REPRIORITIZE: Task reprioritization has finished.</p>
Event	A formatted description of the event.

Task Operations

Task operations provide the ability to override manually the configured task-handling business logic. Task operations are performed on the selected task, as described in “Task Monitoring” on [page 207](#).

Hold Tasks

The Hold Tasks operation holds the selected task, so that no further processing is performed on a task until it is resumed by the Resume Tasks operation. When a task is held, it will not be reprioritized, distributed, or assigned to an agent.

Resume Tasks

The Resume Tasks operation resumes processing of a held task. Only held tasks can be resumed.

Cancel Tasks

The Cancel Tasks operation permanently cancels processing of the selected task. A task cannot be canceled if it has been completed, canceled, or rejected. A task can be canceled if it is already in an Assigned state. The task can be canceled by a source system (through an iWD capture adapter) or by a user through iWD Manager's Global Task List view.

Modify Tasks

The Modify Tasks operation allows an update of a number of task attributes and, optionally a restart of the task. [Figure 68](#) shows an example.

The screenshot shows the 'ACME Solution > Financial Department' interface. At the top, there is a 'Filter' dropdown set to 'All' and an 'Advanced' button. Below this is a table with columns: ID, Status, Media Type, Channel, Process, Created D/T, Business Value, Priority, and Task Due D/T. The table contains four rows of task data. The second row is selected. Below the table are buttons: Refresh, Hold, Resume, Cancel, Modify, Export to XML, and Configuration. The 'Modify Task(s)' dialog box is open, showing 'Core Attributes' and 'Extended Attributes' sections. The 'Core Attributes' section includes fields for Business Value, Priority, Activation D/T, Due D/T, Reprioritization D/T, Expiration D/T, Category, and Channel. The 'Extended Attributes' section includes fields for Customer Segment, Customer ID, Product, Subproduct, TOS Tenant, TOS Process, TOS Subprocess, TOS Created D/T, and TOS Due D/T. At the bottom of the dialog are buttons: Save, Save & Restart, and Cancel.

ID	Status	Media Type	Channel	Process	Created D/T	Business Value	Priority	Task Due D/T
0230PQ2CDMDEG003	Completed	sms		Refund	2010.19.8 15:57	100	150	2010.19.8 16:03
0230PQ2CDMDEG004	Completed	workitem	MQ	Dunning	2010.19.8 15:59	80	130	2010.19.8 16:09
0230PQ2CDMDEG005	Completed	sms		Refund	2010.19.8 16:00	100	100	
0230PO2CDMDEG006	Completed	sms		Refund	2010.19.8 16:00	100	100	

Modify 1 Task(s)

Core Attributes

Business Value	<input type="checkbox"/>	80
Priority	<input type="checkbox"/>	130
Activation D/T	<input type="checkbox"/>	
Due D/T	<input type="checkbox"/>	2010.19.8 16:09
Reprioritization D/T	<input type="checkbox"/>	2010.19.8 16:08
Expiration D/T	<input type="checkbox"/>	2010.19.9 15:59
Category	<input type="checkbox"/>	
Channel	<input type="checkbox"/>	MQ

Extended Attributes

Customer Segment	<input type="checkbox"/>	Gold
Customer ID	<input type="checkbox"/>	Sigma, Inc
Product	<input type="checkbox"/>	Widget1
Subproduct	<input type="checkbox"/>	Cables
TOS Tenant	<input type="checkbox"/>	ACME
TOS Process	<input type="checkbox"/>	Account_Deaktivierung
TOS Subprocess	<input type="checkbox"/>	Delivery
TOS Created D/T	<input type="checkbox"/>	2010.19.8 18:00
TOS Due D/T	<input type="checkbox"/>	2010.19.8 18:00

Figure 68: Modify Task

An attribute will be updated only if the check box that is next to it is checked; it will be checked automatically if a value of the corresponding field has changed. If the task is also restarted, its status is set to *New*, and it is classified and prioritized again in the same way as a new task.

Example—Save & Restart

Consider the following use case: a task is assigned to a Process, but that Process has a start date that is in the future. In this scenario, the task is placed into the *iWD_Rejected* queue. How can this task be re-initiated once the start date of the Process has been reached? In order to re-initiate processing of this task, you must perform a *Save & Restart* in the Global Task List. Processing of the task will not re-initiate automatically.

Select one or more tasks in the Global Task List and click on the *Modify* button. At the bottom of the screen, click *Save & Restart*. You do not have to modify any attributes. The result is that the interaction (task) will be placed back into the *iWD_New* queue in the *iWD* business process.

This *Save & Restart* action might be taken if the task is in the *Rejected* status, or potentially for other business reasons where the task should be treated as if it has just been captured.

Export Tasks to XML

The Export Tasks to XML operation exports all selected tasks from the Global Task List to an XML file. The XML file will contain all of the available attributes for each task in the standard iWD format.

Once it has been exported, a task from an XML file can be imported into a third-party application (such as Microsoft Office Excel) for further analysis and processing.

The setUpdateTrigger Function in URS

The URS setUpdateTrigger function can be used in the Distribution routing strategy prior to the task (interaction) going to the Target block. Then, while the task (interaction) is in the Target block waiting to be routed to an agent, if the Priority attribute is modified through the Global Task List, the internal queue in URS will take this new priority into account. Refer to the *Universal Routing 8.0 Reference Manual* for more information about this function.

Note: The setUpdateTrigger function was introduced in URS 8.0.1.

Filters

Filters allow you to refine the list of tasks that are displayed in the Global Task List. Each filter is defined by a set of filter criteria (optional) and table columns that will be displayed in the Global Task List. [Figure 69](#) shows an example.

Filters > Held_

Name: Held_ ☒ Public

Filter Criteria

Criteria	
Status is NewHeld	[-]
Or	
Status is Held	[-]
Or	
Status is ErrorHeld	[-]

Table Columns

Column	
ID	[-]
Status	[-]
Media Type	[-]
Channel	[-]
Process	[-]
Created D/T	[-]
Business Value	[-]
IWD Priority	[-]
Task Due D/T	[-]

Select criteria to add:

Select column to add:

Save Save & Close Cancel Delete

Figure 69: Filters

Figure 45 lists the attributes and actions that are available in Filters view.

Table 45: Filter Attributes

Attribute/Action	Description
Name	The name of the filter.
Public	Whether the filter will be available to all users (checked) or only the current user (unchecked).
Filter Criteria	Tasks that do not match the defined criteria will be excluded from the Global Task List when the filter is selected. New criteria conditions can be added by selecting them from the <code>Select criteria to add</code> drop-down. Some criteria conditions are parameterized; for such conditions, parameters can be configured directly in a criteria table (such as status for “Status is ...” criteria).
Table Columns	The Global Task List will display these columns when the filter is selected. Columns can be added, removed, and reordered.
Save, Save & Close, Cancel, Delete	Standard iWD Manager functions as described in “iWD Manager Overview” on page 97 .

Search by Attributes that are not Displayed in the Advanced Filter

This section provides some examples how to search by attributes that are not displayed in the Advanced filter by default.

Scenario

You want to filter by an attribute that does not already appear as a column in the `interactions` database table. In this case, the data is being held in BLOB format in the `flexible_properties` column in the database table, but you can add a new column to represent this attribute.

For example, you are submitting tasks through the XML File Capture Adapter, and your messages contain a custom data tag with the key `<myCity>` which is used to send a customer’s city to iWD as part of the task. You want to be able to search for tasks in the Global Task List based on the customer’s city.

Your XML file looks something like this:

```
<GTLMessages>
<CreateTask>
.
<Data>
<myCity>Calgary</myCity>
```



```
</Data>  
</CreateTask>  
</GTLMessages>
```

You need to do two things in order to enable filtering by this custom attribute.

1. Add a new column to the Interaction Server database `interactions` table, giving it a meaningful name that you want to see in the Global Task List's Advanced filter drop-down list, such as `City`. Save the change to the database.
2. In Configuration Manager or Genesys Administrator, add a new Interaction Custom Property. This is a type of Business Attribute. You can find this by going into Configuration Manager or Genesys Administrator. Under your Tenant, go to Business Attributes and look at the Attribute Values of the Interaction Custom Property object. The value of the Name property on the General tab is important; it must be the same as the key of your custom attribute. In this example it would be `myCity`. On the Annex tab, the value of the `translate-to` option in the translation section must exactly match the name of the database column you added in [Step 1](#).
3. Restart Interaction Server.
4. Restart your web application server (such as Tomcat) if it is already running.
5. You should now see your new attribute name in the Advanced filter drop-down list.

Note: Refer to “Task Attributes and Interaction Properties” on [page 245](#) for more information about task attributes and interaction properties that can be used for Filtering in the Global Task List.



Chapter

6

High Availability

This chapter describes high-availability and redundancy in iWD. This chapter is divided into the following sections:

- [Redundancy, page 219](#)
- [Configuring Multiple Business Context Management Services, page 221](#)
- [Deploying Services on Multiple Runtime Nodes, page 222](#)
- [Interaction Server Redundancy, page 223](#)

Redundancy

A redundant service configuration maintains application availability by eliminating all single points of failure within the application itself. iWD provides the option of distributed deployment across logical and physical servers to minimize single points of hardware or OS/application-service software failures. [Figure 70](#) illustrates this configuration.

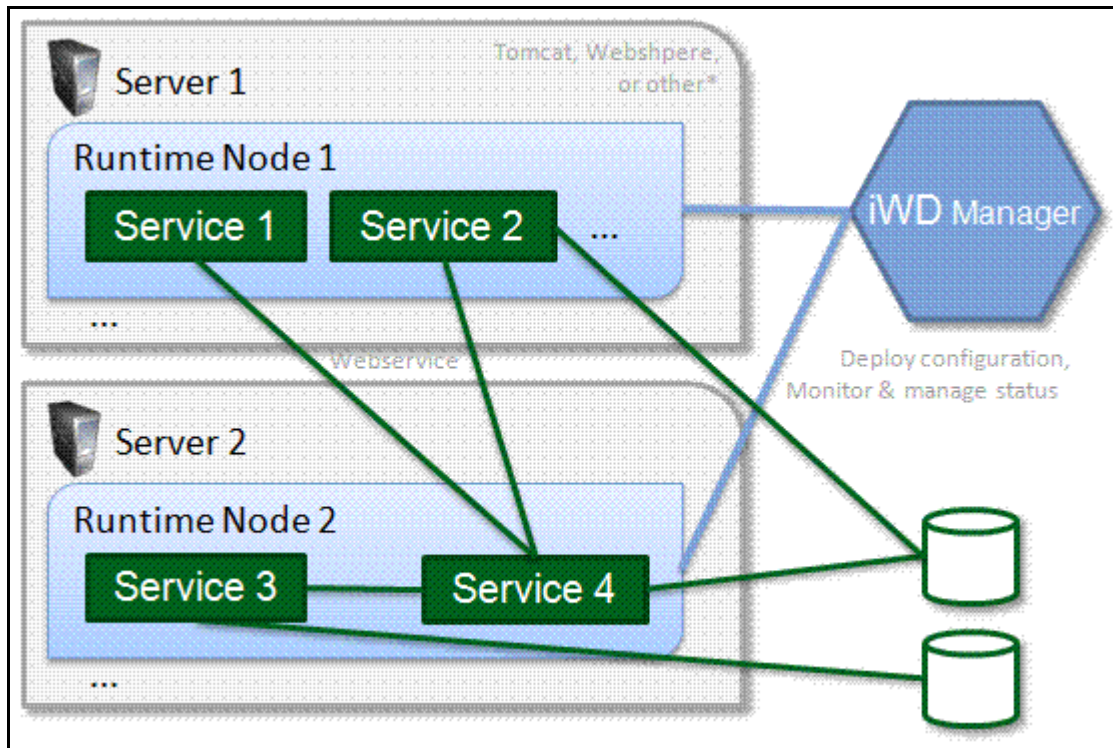


Figure 70: Distributed Deployment with Centralized Management

In addition to this distributed architecture, a redundant solution typically comprises two services:

- A primary service that runs during normal operations
- A backup service that supports the primary service in case of failure, with a failover mechanism that ensures that the backup will take over from the primary service in case of service failure.

High Availability for iWD Components

In iWD 8.0, the iWD Capture Points support hot standby.

Redundancy Support Matrix for iWD Services

[Table 46](#) lists intelligent Workload Distribution services and their supported redundancy mode or modes:

Table 46: Redundancy Support Matrix

iWD Service	Redundancy Support
WebSphere MQ Capture	Hot Standby
XML Capture	Hot Standby

Table 46: Redundancy Support Matrix (Continued)

iWD Service	Redundancy Support
Database Capture	Hot Standby
iWD Data Mart ETL	N/A
Web Service Capture Point	N/A
Business Context Management Service	N/A (configured in the Configuration Management environment)

Configuring Multiple Business Context Management Services

It is possible to configure multiple Business Context Management Services (BCMS) for purposes of high availability or load balancing.

It is important to remember that from a Genesys Configuration Server perspective, the BCMS is a “Third Party Server” to which Interaction Server establishes a connection in order to process requests that will be passed to the iWD rule engine. Interaction Server manages the connection to the BCMS, and how it passes on requests to the BCMS depends on several factors, which will differ depending on the goal of the configuration. The first option is a primary/backup configuration. The second option is a load balancing configuration.

Primary/Backup Configuration

You may configure multiple BCMS in a primary/backup mode. On the BCMS applications themselves (the applications defined in Configuration Server, not the services configured through iWD Manager), you will configure one as primary and one as backup. They can be on the same host or on different hosts, but the ports must be different. On the **Connections** tab of the Interaction Server application you will only specify the *primary* BCMS.

On the **External Service** blocks in your routing strategies, where the BCMS external service is invoked, you should list the name of your BCMS application that is serving as the primary. See also “Business Context Management Service” on [page 168](#) and “Configuration of List Objects” on [page 252](#).

You will also need to configure two BCMS services in iWD, each of which will correspond to one of the applications defined in Configuration Server.

Load Balancing Configuration

In this scenario, you do not configure the BCMS applications as primary/backup. Configure your two BCMS applications as an “application cluster”. You do this by creating a new Third Party application in Configuration Server of type `ApplicationCluster` (the application template is on the Interaction Management CD and must be imported first). When you create the application, give it a meaningful name such as `BCMS_Cluster_1`. Because you need to define different BCMS applications for each Solution in iWD, it is recommended to number the application clusters or use some other nomenclature so you can understand the correlation. In the `Connections` tab of this `ApplicationCluster` object you will add connections to your **two** (or more) BCMS applications. In the `Connections` tab of your Interaction Server, add a connection to this `ApplicationCluster`.

Now, in the List Object (see “Configuration of List Objects” on [page 252](#)) where the correspondence between the BCMS application names and the iWD Solution IDs is defined, instead of using an actual BCMS application name, you will use the name of your application cluster, such as `BCMS_Cluster_1`.

You will not have to change the default logic in the IWDBP business process, because the routing strategies will use that List Object to retrieve the name of the ESP application based on the iWD Solution ID. In this case, it will retrieve the name of the `ApplicationCluster` application, which will instruct Interaction Server to use the two BCMS applications under that cluster when making requests to the external service.

Note: Refer to the *eServices (Multimedia) 8.0 User's Guide* for more information about using Application Clusters for high availability of ESP Servers.

ESP Object in Interaction Routing Designer

If you do not have the option in IRD to select an application cluster in the ESP object, you can update the options for your IRD application. In the `default` section, add an option named `tools-tuneup` and set the value to `extended`. Restart IRD.

Deploying Services on Multiple Runtime Nodes

A Solution's services can be deployed on multiple runtime nodes. For example:

- “Backup” services should be deployed on a separate runtime node from “Primary” services.

- Data Mart-related services should be deployed on yet another runtime node, separate from the backup and primary services

The following procedure provides deployment steps for installing and configuring multiple Runtime Nodes.

Procedure: Deploying Services on multiple Runtime Nodes

Purpose: To install and configure multiple Runtime Nodes in your environment.

Start of procedure

1. Install additional Application Server(s) (for example, additional instances of Tomcat).
2. Run the iWD Runtime Node setup on every Application Server.
3. Add additional Runtime Node application(s) in Configuration Manager or Genesys Administrator that map to each installed Runtime Node.
4. Add Runtime Node(s) in iWD Manager for each installed Runtime Node.
5. Install the following components, specifying the specific iWD Runtime Node to which you are deploying them:
 - iWD Rules
 - iWD Capture
 - iWD Data Mart
6. When deploying for backup purposes rather than partitioning purposes, you must run the install of iWD Rules and iWD Capture twice; once against each Runtime Node (you cannot install iWD Data Mart against two Runtime Nodes).

End of procedure

Next Steps

- Configure iWD Services in iWD Manager, specifying the desired Runtime Node.

Interaction Server Redundancy

Refer to the *eServices 8.0 User's Guide* for information about high-availability and redundancy supported by Interaction Server.

A

iWD Messages

This Appendix describes the messages that are supported in iWD 8.0.

It contains the following sections:

- [Messages, page 225](#)
- [Task Attributes and Interaction Properties, page 245](#)

Messages

This section provides a detailed description of all of the input and output iWD Messages that are supported. The following information is documented for each message:

- Direction—“In” or “Out”. All “In” messages come 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 47 on [page 226](#)) 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 only applicable 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 only applicable for messages that have an “In” direction.

This Appendix also includes the section “Task Attributes and Interaction Properties” on [page 245](#), which explains how task attributes are mapped to interaction properties.

Note: There is now an XML schema available for the iWD messages. It is available here:

<https://sites.google.com/a/iwdlab.com/iwd8/capture/schema>

Data Types

[Table 47](#) describes the data types used in iWD messages.

Table 47: Data Types for iWD Messages

Type	Description
Integer	An integer value ($-2^{31} < \text{value} < 2^{31}$).
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 Capture Point.

Task Action

Direction: In

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

Format:

```
<GTLMessages>
  <ActionName>
    <BrokerId>String</BrokerId> 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 48](#).

Table 48: Attributes for Task Action Messages

Attribute	Description
BrokerId	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 230](#)).

Error codes: See [Table 49](#).

Table 49: 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 BrokerId is not found. This error code can be triggered for all action messages, except for the CreateTask message (see “CreateTask” on page 228).

Task Notification

Direction: Out

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

Format:

```
<GTLMessages>
  <[notification name]>
    <BrokerId>String</BrokerId>
    <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 50](#).

Table 50: Attributes for Task Notification Messages

Attribute	Description
BrokerId	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.

Note: In the `<CreateTask>` message for the Web Service Capture Point, the iWD core attribute `createdDateTime` is now mandatory, not optional. The value can be null, however. The reason it has been changed from optional to mandatory is to support migration of tasks from iWD 7.6.1 to 8.0, where `createdDateTime` would be populated during the redistribution of tasks from 7.6.1 to 8.0.

Format:

```

<GTLMessages>
  <CreateTask>

```

Standard action attributes, as documented in “Task Action” on [page 226](#), except for `BrokerId`.

```

    <channel>String(32)</channel>
    <mediaType> string(32) </mediaType>
    <category>String(32)</category>
    <activationDateTime>DateTime</activationDateTime>
    <dueDateTime> DateTime </dueDateTime>
    <expirationDateTime> DateTime </expirationDateTime>
    <businessValue> Integer </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 51](#).

Table 51: 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 <code>NewHeld</code> and will not be processed further, until a subsequent <code>ResumeTask</code> message (see “ ResumeTask ” on page 241).
CaptureId (optional)	If a <code>CaptureId</code> is not provided, it will be assigned to the same generated value as <code>BrokerId</code> .
mediaType	If a media type is specified, it will override the default media type configured for the iWD Capture Point Service that is being used.

See “Task Action” on [page 226](#) and “TaskInfo” on [page 231](#) for the description of the remaining attributes.

Response message: `TaskCreated` (see “`TaskCreated`” on [page 230](#)).

Error codes: see [Table 52](#).

Table 52: 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 228](#)) and indicates successful task creation.

Format:

```
<GTLMessages>
  <TaskCreated>
    Standard notification attributes, as documented in “Task Notification”
    on page 227.
  </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 53](#).

Table 53: 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: Request task details by the given task's capture ID or interaction ID.

Format:

```
<GTLMessages>
  <GetTaskInfo>
    Standard action attributes, as documented in "Task Action" on page 226.
  </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 231](#)) and provides detailed information about the requested task.

Format:

```
<GTLMessages>
  <TaskInfo>
    Standard action attributes, as documented in "Task Action" on
    page 226.
    <tenantId>String(16)</tenantId>
    <solutionId> String(16)</solutionId>
    <DepartmentId> String(16)</DepartmentId>
    <processId> String(16)</processId>
    <channel>String(32)</channel>
    <mediaType> String(32) </mediaType>
    <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>
    <distributionPointId>String(16)</distributionPointId>
    <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>
    <requestedAgent>String(64)</requestedAgent>
    <requestedSkill>String(64)</requestedSkill>
    <requestedAgentGroup>String(64)</requestedAgentGroup>
    <requestedPlaceGroup>String(64)</requestedPlaceGroup>
    </Ext>
    <Data>
        <customAttribute1> String(255)</customAttribute1>
        ...
    </Data>
    <TaskInfo>
    <GTLMessages>

```

Attributes: See [Table 54](#).

Table 54: 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 in 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 in the CIM Platform are ignored.

Table 54: Attributes for TaskInfo Messages (Continued)

Attribute	Description
DepartmentId	The task'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 <code>IWD_DepartmentId</code> key; updates in the CIM Platform are ignored.
processId	The task'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 <code>IWD_processId</code> key; updates in 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 <code>IWD_channel</code> key; updates in the CIM Platform are picked up.
category	The task's category—for example Followup. This attribute is submitted to the CIM Platform with the <code>IWD_category</code> key; updates in the CIM Platform are picked up.
status	<p>Task status:</p> <p>New—The task has just been created and will be processed.</p> <p>Captured—The task was processed, but it has not been prioritized.</p> <p>Queued—The task was processed and prioritized at least once.</p> <p>Assigned—The task is assigned to an agent.</p> <p>Completed—The task is completed.</p> <p>Held—The task is held and will not be reprioritized or distributed until it is resumed.</p> <p>Error—An error has occurred during task processing, prioritization, or distribution. Error details are stored in the custom extended task attribute <code>Error</code>. The task can be restarted, and iWD will attempt to process the task again.</p> <p>Canceled—The task is canceled.</p> <p>Rejected—The task was rejected during processing. This can occur when the task is assigned to an expired department or process.</p>
businessCalendarId	<p>The ID of the business calendar that is assigned to the task, as configured in iWD Manager.</p> <p>Note: Maximum length is 16 characters</p>
createdDateTime	The date/time when the task has been created in iWD. This attribute is submitted to the CIM Platform with the <code>ReceivedAt</code> key; updates in the CIM Platform are ignored.

Table 54: Attributes for TaskInfo Messages (Continued)

Attribute	Description
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.
priority	The task priority, which is an integer value that is used to order tasks. The higher the value, the higher the task will stand in the queue and the sooner 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

Table 54: Attributes for TaskInfo Messages (Continued)

Attribute	Description
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.
distributionPointId	This attribute is provided for backward compatibility.
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.
resultCode	The task result code/outcome; typically, 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_sourceFirstCreatedDateTime 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.

Table 54: Attributes for TaskInfo Messages (Continued)

Attribute	Description
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.
requestedAgent	The agent requested for the task.
requestedSkill	The skill requested for the task.
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 BrokerId 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 226.
    <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>
  </UpdateTask>
</GTLMessages>

```

```

        <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 226](#) and “TaskInfo” on [page 231](#) for a description of the attributes.

Response message: “TaskUpdated”.

Note: If you use UpdateTask to update a task’s priority, in order for that updated priority to be taken into account in your routing strategy you may want to use the URS function `setUpdateTrigger`. By using this URS function, if the `Priority` attribute of a task is updated—even while the task is in a Target block of a routing strategy waiting to be routed to an agent—the internal queue in URS will take this new priority into account. For more information about this function, refer to the *Universal Routing 8.0 Reference Manual*.

The `setUpdateTrigger` function was introduced in URS 8.0.1.

TaskUpdated

Direction: Out

Description: The TaskUpdated message is submitted as a response to the UpdateTask message (see “UpdateTask” on [page 236](#)), 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 227](#).

```

    <tenantId>String(16)</tenantId>
    <solutionId> String(16)</solutionId>
    <DepartmentId> String(16)</DepartmentId>
    <processId> String(16)</processId>
    <channel>String(32)</channel>
    <mediaType> String(32) </mediaType>
    <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 226](#) and “TaskInfo” on [page 231](#) 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 reserved for iWD.

Format:

```

<GTLMessages>
    <TaskDistributedQueue>
        Standard notification attributes, as documented in “Task Notification”
on page 227.
        <Queue>String(255)</Queue>
        <QueueType>String(16)</QueueType>
        <QueueTarget>String(255)</QueueTarget>
    </TaskDistributedQueue>
</GTLMessages>

```

Attributes: See “Task Notification” on [page 227](#) and “TaskInfo” on [page 231](#) 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 227.
    <AssignedToUser>String(64)</AssignedToUser>
  </TaskAssigned>
</GTLMessages>
```

Attributes: See “Task Notification” on [page 227](#) and “TaskInfo” on [page 231](#) 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 226.
  </CompleteTask>
</GTLMessages>
```

Attributes: See “Task Action” on [page 226](#) for a description of the attributes.

Response message: “TaskCompleted”.

Error codes: See [Table 55](#).

Table 55: Error Codes for CompleteTask Messages

Error code	Description
CANNOT_COMPLETE_TASK	Cannot complete the task, because it is already completed or assigned.

TaskCompleted

Direction: Out

Description: The TaskCompleted message is submitted as a response to the CompleteTask message (see “CompleteTask” on [page 239](#)), 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 227](#)

```
</TaskCompleted>
</GTLMessages>
```

Attributes: See “Task Notification” on [page 227](#) 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 241](#)).

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 226.
  </HoldTask>
</GTLMessages>
```

Attributes: See “Task Action” on [page 226](#) for a description of the attributes.

Response message: “TaskHeld”.

Error codes: See [Table 56](#).

Table 56: Error Codes for HoldTask Messages

Error code	Description
CANNOT_HOLD_ASSIGNED_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 240](#)), 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 227.
  </TaskHeld>
</GTLMessages>
```

Attributes: See “Task Notification” on [page 227](#) for a description of the attributes.

TaskErrorHeld

Direction: Out

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

Format:

```
<GTLMessages>
  <TaskErrorHeld>
    Standard notification attributes, as documented in “Task Notification”
    on page 227.
    <Error>String(255)</Error>
  </TaskErrorHeld>
</GTLMessages>
```

Attributes: See “Task Notification” on [page 227](#) 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 226.
  </ResumeTask>
</GTLMessages>
```

Attributes: See “Task Action” on [page 226](#) for a description of the attributes.

Response message: “TaskResumed”.

Error codes: See [Table 57](#).

Table 57: 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 241](#)), as well as when a task is held from iWD Manager.

Format:

```
<GTLMessages>
```

```
  <TaskResumed>
```

Standard notification attributes, as documented in “Task Notification” on [page 227](#).

```
  </TaskResumed>
```

```
</GTLMessages>
```

Attributes: See “Task Notification” on [page 227](#) 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.

Format:

```
<GTLMessages>
```

```
  <RestartTask>
```

Standard action attributes, as documented in “Task Action” on [page 226](#).

```
  </RestartTask>
```

```
</GTLMessages>
```

Attributes: See “Task Action” on [page 226](#) for a description of the attributes.

Response message: “TaskRestarted”.

Error codes: See [Table 58](#).

Table 58: Error Codes for RestartTask Messages

Error code	Description
CANNOT_RESTART_TASK	Cannot restart the task, because it is held or assigned.

TaskRestarted

Direction: Out

Description: The TaskRestarted message is submitted as a response to the RestartTask message (see “RestartTask”, above), 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 227](#).

```
  </TaskRestarted>
```

```
</GTLMessages>
```

Attributes: See “Task Notification” on [page 227](#) 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 226.
  </CancelTask>
</GTLMessages>
```

Attributes: See “Task Action” on [page 226](#) for a description of the attributes.

Response message: “TaskCanceled”.

Error codes: See [Table 59](#).

Table 59: Error Codes for CancelTask Messages

Error code	Description
CANNOT_CANCEL_TASK	Tasks that have already been canceled cannot be canceled again, and tasks that have been assigned cannot be canceled. All other tasks can be 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 227.
  </TaskCanceled>
</GTLMessages>
```

Attributes: See “Task Notification” on [page 227](#) for a description of the attributes.

TaskRejected

Direction: Out

Description: The TaskRejected message is submitted when the task gets rejected by the iWD Classification Service. The task can be rejected when a process or department to which the task gets 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 227.
  </TaskRejected>
</GTLMessages>
```

Attributes: See “Task Notification” on [page 227](#) for a description of the attributes.

Ping

Direction: In

Description: A simple Ping message that can be used to check the health of the 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 244](#)), indicating that the Capture Point service is active. The Pong message contains the ID that was sent in the Ping message.

Format:

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

Task Attributes and Interaction Properties

iWD task attributes are separated into three categories:

- Core
- Extended
- Custom

Most of the iWD task attributes can be set when a task is created or updated through an iWD capture adapter, although some, such as interaction ID, are set by iWD components or by Interaction Server. Most of the task attributes are displayed in the `Attributes` tab when a task is selected in the Global Task List.

These attributes are maintained as pieces of attached data of the interaction, as it is stored in the Interaction Server's interactions database table. Some of the attributes are stored in independent columns in that database table, while others are stored in a binary (BLOB) format in a column in the interactions table called `flexible_properties`.

There are many reasons to update or access the data stored in the iWD task attributes, including:

- Setting the value of one or more task attributes as part of an iWD message such as `CreateTask`, when working with an iWD capture adapter.
- Reading or updating task attributes in business rules.
- Using the data in the `Condition`, `Order`, and `Segmentation` tabs of `Views` in Genesys Business Processes.

Note: You cannot use properties with a `Timestamp` data type on the `Segmentation` tab.

- Reading or updating the data contained in task attributes within a routing strategy.
- Making the data available to an agent or knowledge worker desktop application, either to display to the agent or to facilitate a screen pop.
- Filtering the display of the Global Task List.

Note: There are specific columns in the `interactions` table that you should **not** change. Please refer to Chapter 8, "Interaction Properties", in the *eServices 8.0 User's Guide*.

[Table 60](#) provides information about how the iWD task attributes map to the columns in which their values are stored in the interactions database table, as well as the key name that is used in the attached data of the interaction, and the label used for that attribute as it appears on the `Attributes` tab of the Global Task List.

Refer to the beginning of this Appendix for details about how these task attributes are used in iWD messages such as `CreateTask`.

Table 60: Task Attribute Mapping

Task Attribute	Column Name in Interactions Table	Type	Interaction Attached Data Key	Label in Global Task List
Core Attributes				
BrokerId	Id	String	InteractionId	ID
Status ^a	queue	String	Queue	Status
mediaType	media_type	String	MediaType	Media Type
TenantId	IWD_tenantId	String	IWD_tenantId	Tenant
businessCalendarId	in <flexible_properties>	String	IWD_businessCalendarId	Business Calendar
DepartmentId	IWD_departmentId	String	IWD_departmentId	Department (name is shown instead of ID)
ProcessId	IWD_processId	String	IWD_processId	Process (name is shown instead of ID)
Channel	IWD_channel	String	IWD_channel	Channel
Category	IWD_category	String	IWD_Category	Category
CapturePointID	IWD_capturePointId	String	IWD_capturePointId	Capture Point (value is shown instead of ID)
CaptureId	external_id	String	ExternalId	Capture ID
CreatedDateTime	received_at	Timestamp	ReceivedAt	Capture D/T
DistributionPointId	IWD_distributionPointId	String	IWD_distributionPointId	Distribution Point
ActivationDateTime	IWD_activationDateTime	Timestamp		Activation D/T
BusinessValue	IWD_businessValue	String	IWD_businessValue	Business Value
DueDateTime	IWD_dueDateTime	Timestamp	IWD_dueDateTime	Task Due D/T
Priority	priority	Integer	Priority	Priority

Table 60: Task Attribute Mapping (Continued)

Task Attribute	Column Name in Interactions Table	Type	Interaction Attached Data Key	Label in Global Task List
ReprioritizeDate Time	IWD_reprioritizeDateTime	Timestamp	IWD_reprioritize DateTime	Reprioritization D/T
AssignedToUser	assigned_to	String	RTargetAgent Selected	Assigned To
AssignedDateTime	assigned_at	Timestamp	AssignedAt	Assigned D/T
-	completed_at	Timestamp	CompletedAt	Completed D/T
ExpirationDateTime	IWD_expirationDateTime	Timestamp	IWD_expiration DateTime	Expiration D/T
-	IWD_solutionId	String	IWD_solutionId	-
Extended Attributes				
CustomerSegment	IWD_ext_customer Segment	String	IWD_ext_customer Segment	Customer Segment
CustomerId	IWD_ext_customerId	String	IWD_ext_customer Id	Customer ID
ProductType	IWD_ext_productType	String	IWD_ext_product Type	Product
ProductSubtype	IWD_ext_sourceProduct Subtype	String	IWD_ext_product Subtype	Subproduct
RequestedAgent Group	IWD_ext_RequestedAgent Group		IWD_ext_ RequestedAgent Group	Requested Agent Group
RequestedPlace Group	IWD_ext_RequestedPlace Group		IWD_ext_ RequestedPlace Group	Requested Place Group
SourceTenant	IWD_ext_sourceTenant	String	IWD_ext_source Tenant	TOS Tenant
SourceProcessType	IWD_ext_sourceProcess Type	String	IWD_ext_source ProcessType	TOS Process
SourceProcess Subtype	IWD_ext_sourceProcess SubType	String	IWD_ext_source ProcessSubtype	TOS Subprocess

Table 60: Task Attribute Mapping (Continued)

Task Attribute	Column Name in Interactions Table	Type	Interaction Attached Data Key	Label in Global Task List
SourceFirstCreatedDateTime	IWD_ext_sourceFirstCreatedDT	Timestamp	IWD_ext_sourceFirstCreatedDate Time	-
SourceCreatedDate Time	IWD_ext_sourceCreated DateTime	Timestamp	IWD_ext_source CreatedDateTime	TOS Created D/T
SourceDueDate Time	IWD_ext_sourceDueDate Time	Timestamp	IWD_ext_source DueDateTime	TOS Due D/T
ResultCode	IWD_ext_resultCode	String	IWD_ext_Result Code	Result Code
RequestedAgent	IWD_ext_requestedAgent	String	IWD_ext_requested Agent	Requested Agent
RequestedSkill	IWD_ext_requestedSkill	String	IWD_ext_requested Skill	Requested Skill
Custom Attributes				
myCustomAttribute	in <flexible_properties>	String	myCustomAttribute	myCustom Attribute
ArchiveDestination	in <flexible_properties>	String	ArchiveDestination	Archive Destination
ESP_Result	in <flexible_properties>	String	ESP_Result	ESP_Result
ESP_Error	in <flexible_properties>	String	ESP_Error	ESP_Error

- a. The value of `Status` does not correlate directly to the contents of the `queue` column in the interactions table. It is dynamically calculated, taking into account information such as the queue and whether the task is held or not. Because the contents of the `Status` column are dynamically calculated, rather than being read from a database table, the Global Task List cannot be sorted by the `Status` column. You should use filters instead, if you are interested in focusing in on the contents of the list by this criterion.

Interaction Custom Properties

If you want to use the value of a task attribute in the `Condition`, `Order`, and `Segmentation` tabs of Views in Genesys Business Processes, or if you want to filter or sort the display of the Global Task List by using a task attribute, that task attribute must be represented in an independent column in the Interaction Server's interactions database table. If that task attribute is inside the binary data in the `flexible_properties` column, you must create an Interaction

Custom Property that corresponds to that attribute. The data type of the property can be a timestamp, string, or number.

Note: Properties with the Timestamp data type cannot be used on the Segmentation tab of Views in a Genesys Business Process.

Procedure: Configuring a custom interaction property

Start of procedure

1. Decide on an attached data key that will be the source of the content of the custom property.
2. Create a new field directly in the `interactions` database.
3. Create a new Business Attribute:
 - Name = `InteractionCustomProperties`
 - Display name = `Interaction Custom Properties`
 - Type = `Custom`

If such an attribute already exists go to the next step.

4. Expand `Interaction Custom Properties` and open its `Attribute values`.
5. Give it an `Attribute Value`, with a name exactly matching the attached data key name that you decided on in Step 1. The matching is case sensitive (you can create a separate display name).
6. In your new attribute value, go to the `Annex` tab and create a section called `translation`.
7. In the new `translation` section, create an option called `translate-to`, with its value duplicating the name of the new field you created in [Step 2](#).

End of procedure

Note: If you specify a custom field as not null, you must ensure that you provide some data to that field upon creation of a task. If no data is provided, the request will fail because Interaction Server sends NULL for empty fields, and that will be rejected by the DBMS.

B

iWD Business Process (IWDBP)

This Appendix describes the default iWD business process (IWDBP) that is supplied in the iWD Setup Utility component.

- The *Universal Routing 8.0 Deployment Guide* describes how to have the Interaction Design shortcut bar appear in IRD, if it has not appeared automatically.
- The *Universal Routing 8.0 Business Process User's Guide* provides an in-depth discussion of business processes.
- The *Universal Routing 8.0 Interaction Routing Designer Help* describes how to create, save, import and export a business process, and how to load the strategies that comprise the business process.

Note: When Interaction Routing Designer (IRD) starts up, it checks for an eServices solution installed by the eServices Configuration Wizard. If none is found, the IRD main window does not contain an Interaction Design shortcut bar. You cannot navigate to the Business Processes list pane or open the Interaction Design window. To change the default, use the Views tab in Routing Design Options, which opens from the Tools menu. Clear the default check box and click OK.

This Appendix contains the following sections:

- [Configuration of List Objects, page 252](#)
- [iWD Business Process, page 253](#)
- [Modifying the iWD Business Process, page 272](#)

Configuration of List Objects

One Business Process can serve several solutions under the same tenant. At least one ESP service server has to be configured per solution (see “Configuring Multiple Business Context Management Services” on [page 221](#) for information about using Application Clusters). Capture Points add Solution ID to an interaction automatically. Inside of the business process, an ESP service is invoked. In order to specify which ESP service is invoked, the correlation between an iWD solution and an ESP service is set up inside of a List Object.

The iWD Setup Utility creates an empty List Object called `Iwd_Esp_List`. In environments with only one solution, no further configuration needs to be done on the List Object. If you have multiple solutions (or add one at a later time) the `Iwd_Esp_List` List Object needs to be updated.

Notes: If you run the iWD Setup Utility more than once, the new Business Context Management Service(s) will be added to the existing `Iwd_Esp_List` List Object.

Refer to [page 221](#) for information about configuring multiple Business Context Management Services.

The List Object looks like a list of pairs:

<code>Solution_1</code>	<code>ESPService_1</code>
<code>Solution_2</code>	<code>ESPService_2</code>
<code>Solution_i</code>	<code>ESPService_i</code>

Where the Solution ID is the key, and the name of the Business Context Management Service Application is the value.

Note: It is very important that the pairs are set up correctly. If, for example, `Solution_1` is mapped to `ESPService_2` instead of to `ESPService_1`, business rules for `Solution_2` will be applied to all interactions which were submitted by Capture Points from `Solution_1`.

These key-value pairs in a List Object need to be set up only once per tenant, and can be configured in Interaction Routing Designer (IRD). See [Figure 71](#).

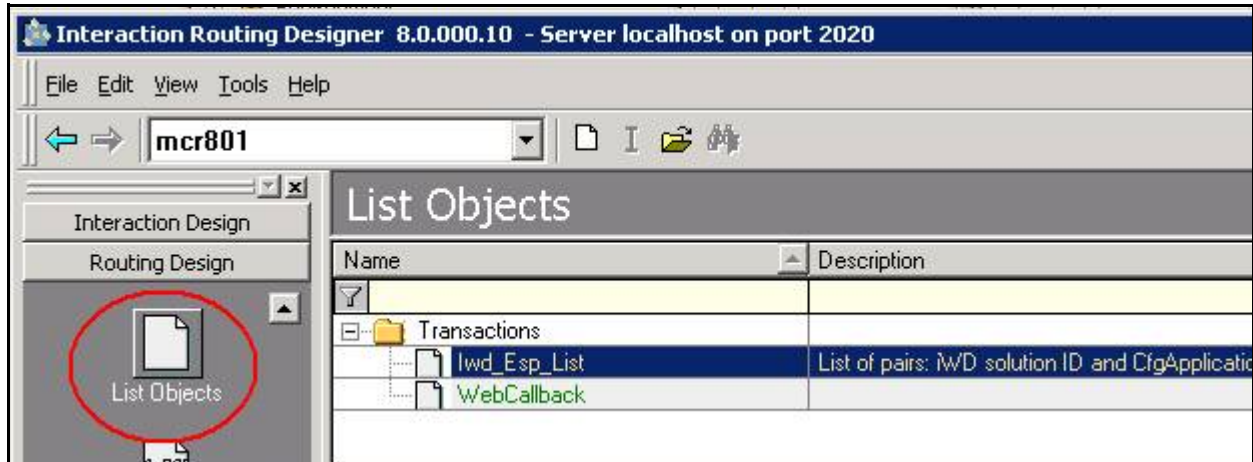


Figure 71: List Objects in IRD

[Figure 72](#) shows the details of the List Object.

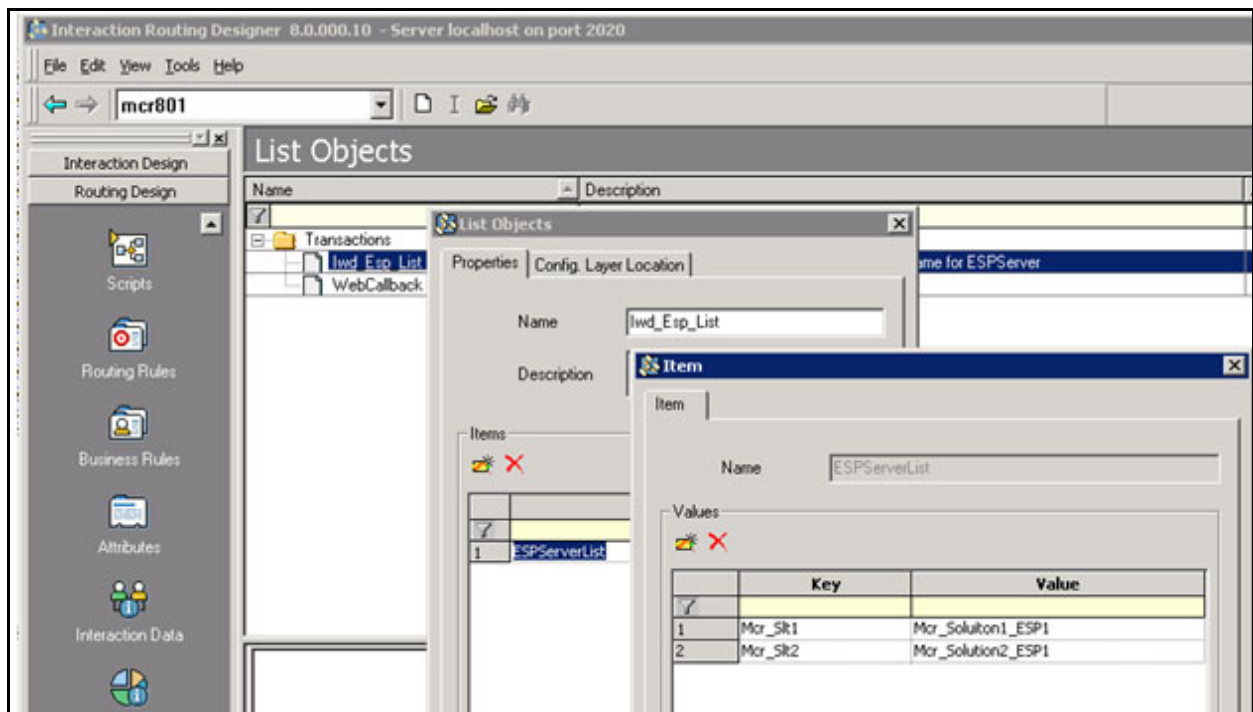


Figure 72: List Object Details

iWD Business Process

The iWD business process (IWDBP) contains the following strategies:

- Classification
- Prioritization

- Distribution
- Archive

The iWD business process contains the following subroutine:

- DetermineESPServerName

The iWD business process contains the following queues:

- iWD_New
- iWD_Captured
- iWD_Queued
- iWD_Canceled
- iWD_Rejected
- iWD_Completed
- iWD_ErrorHeld

The Interaction Queues that are included in the out of the box IWDBP business process must be present, and the names should **not** be changed. The Global Task List looks for specific Interaction Queue names, as they appear in the business process (such as iWD_New and iWD_Queued). If you modify the business process to add additional queues or rename existing queues, the interactions will not display properly in the Global Task List. No status will be displayed for interactions in Interaction Queues with names that the Global Task List does not recognize.

Figure 73 shows the entire business process as it appears in the Interaction Design window of Interaction Routing Designer.

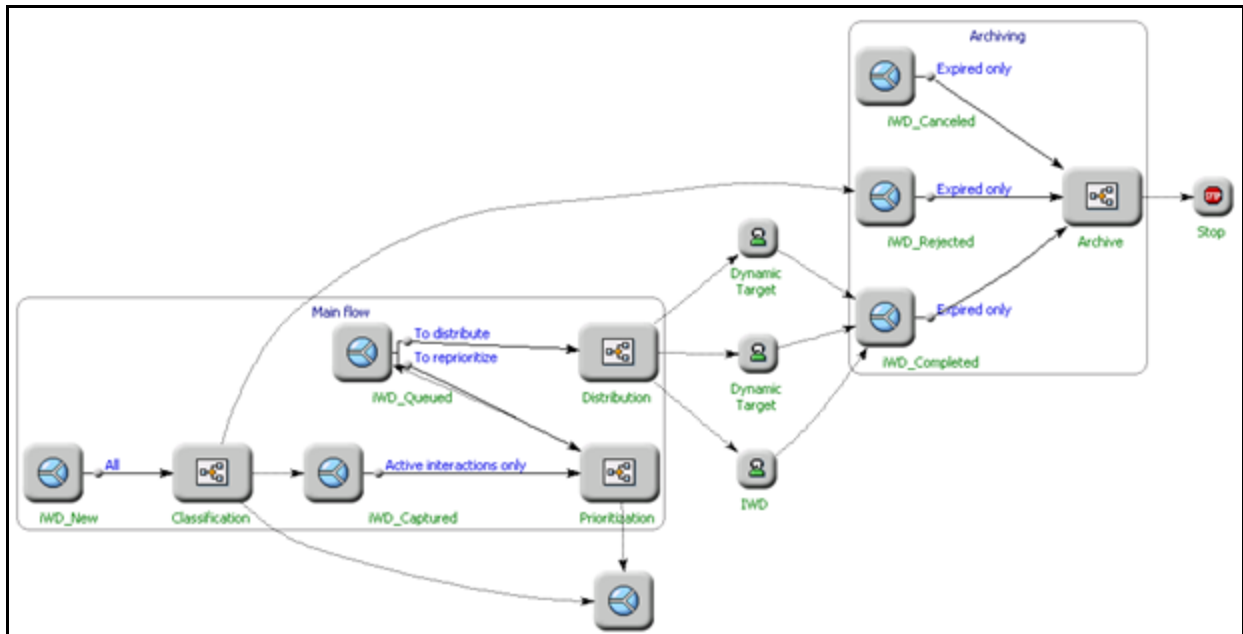


Figure 73: iWD Business Process

The group of objects on the left-hand side are part of the “Main Flow” of the business process. Figure 74 shows this section in more detail. The group of

objects on the right-hand side represent the “Archiving” section of the business process. [Figure 75](#) shows the Archiving section in more detail.

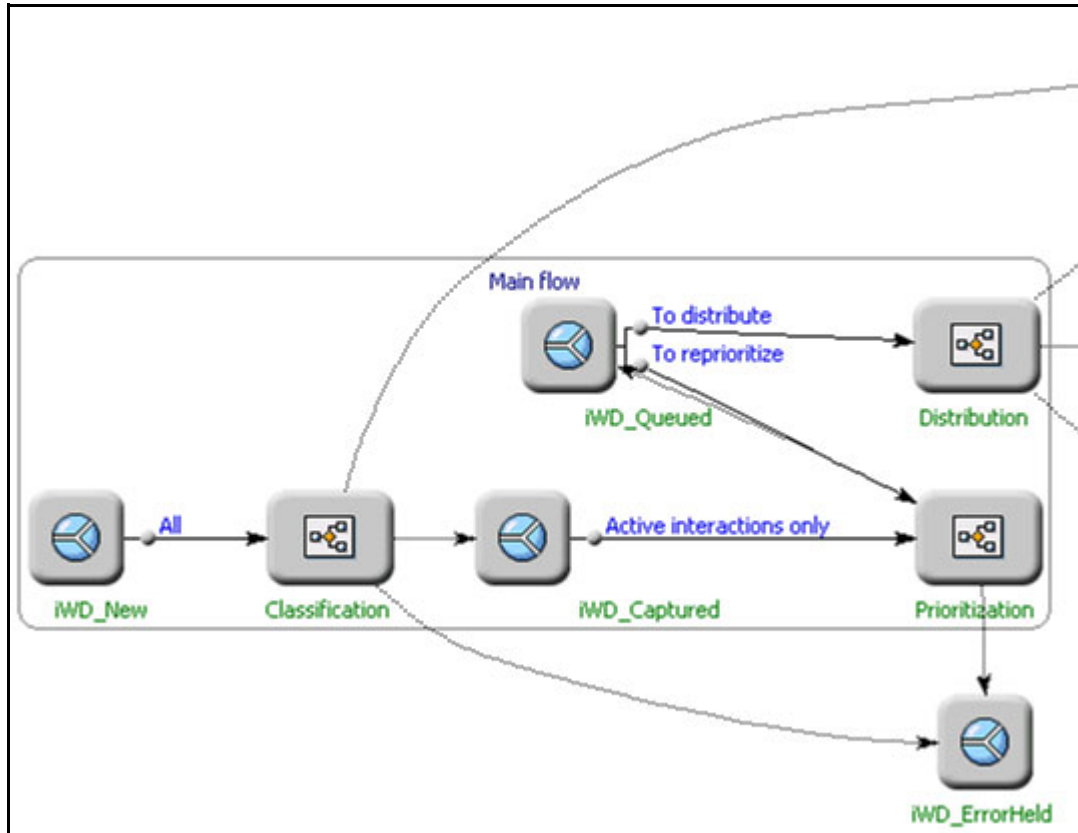


Figure 74: iWD Business Process—Main Flow

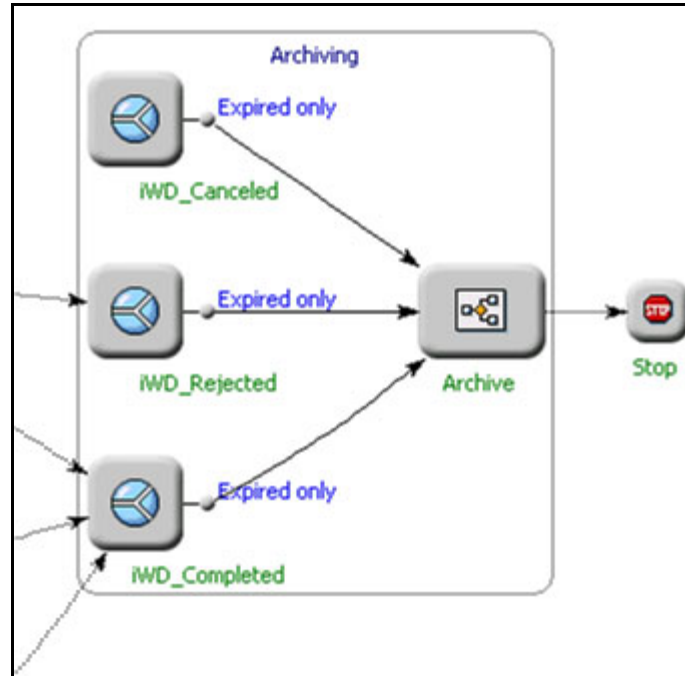


Figure 75: iWD Business Process—Archiving

Classification Strategy

The purpose of this strategy is to invoke corresponding classification rules, analyze the result of the rules application and place the interaction into the appropriate queue, depending on the result.

This strategy processes interactions from the following queues:

- iWD_New

Interactions have to satisfy the following conditions (see [Figure 76](#)):

- There are no conditions here.
- Interactions are taken in order they were submitted.

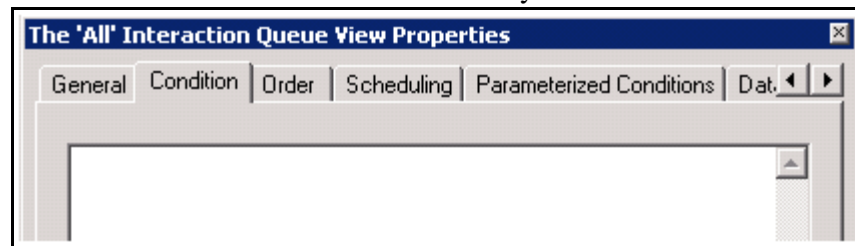


Figure 76: The 'All' Interaction Queue View Properties

Note: ESP stands for External Service Protocol. In this document it is the Business Context Management Service.

Figure 77 shows the Classification strategy.

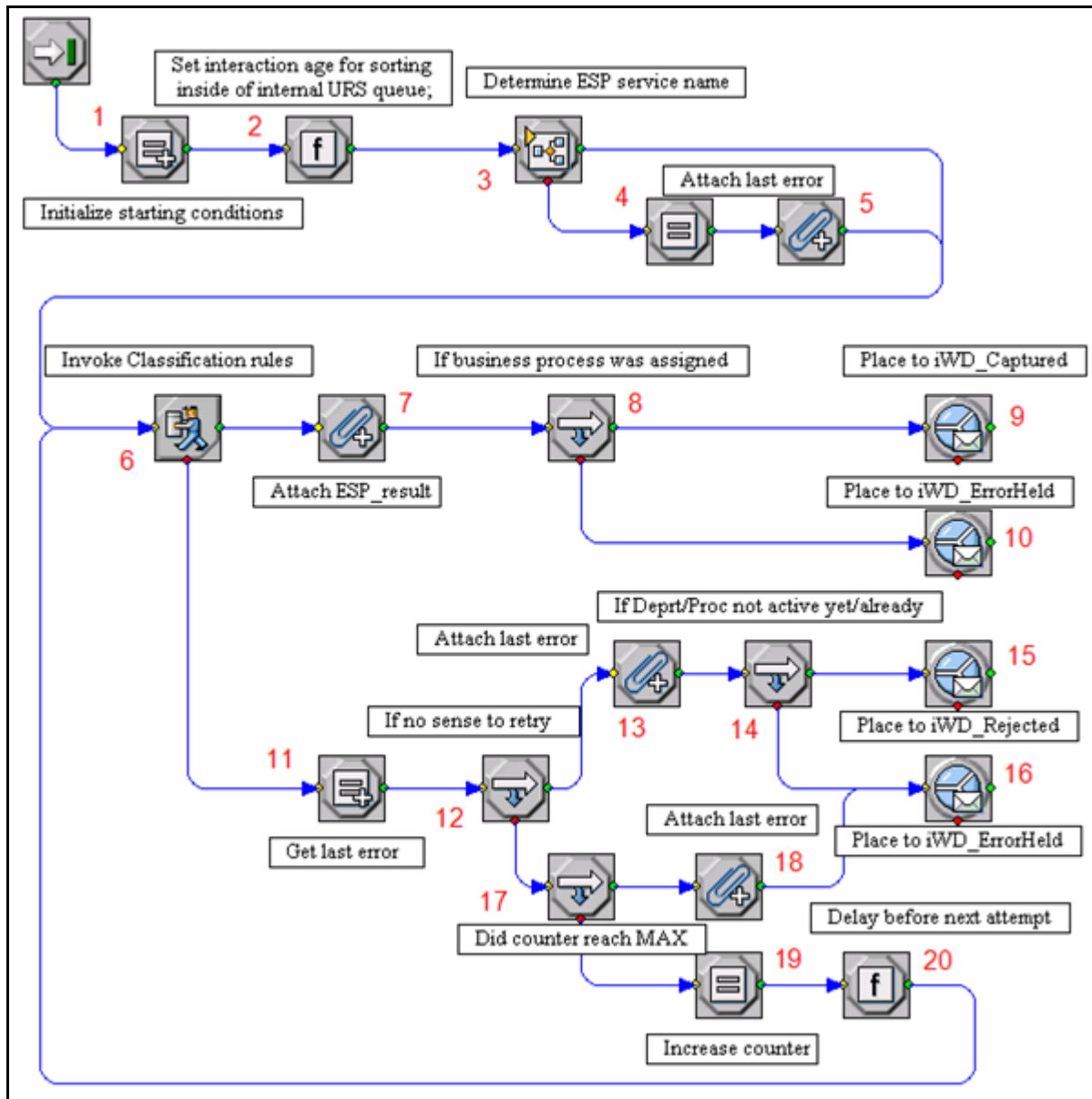


Figure 77: Classification Strategy

Summary of flow

Note: The IRD objects described in this section are keyed to the numbers in Figure 77.

1. Variables are initialized (see Figure 78):
 - `_counter` is the running counter for the numbers of attempts to invoke the ESP service.

- `_max_repeate_counter` specifies how many times to attempt to invoke the ESP service
- `_delay` specifies the delay (in milliseconds) between attempts to invoke rules.

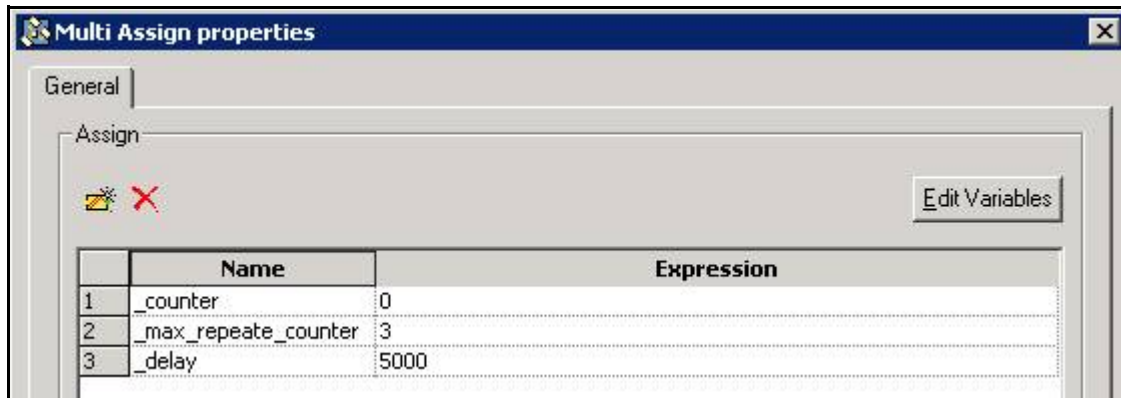


Figure 78: Initialize Variables

2. A command is sent to URS to use interaction age while sorting interactions in internal queues.
3. The subroutine that determines the correct ESP server name is invoked. This subroutine also sets up cases when there is reason to retry to invoke the ESP server.
4. If the subroutine fails an error is extracted.
5. This error is attached to user data as a key-value pair with the key `IWD_ESP_Determination_Error`.
6. A request to apply classification rules to the interaction is sent.
7. If ESP server reports that operation was completed successfully, the results are attached to user data as a key-value pair with the key `ESP_Result`. This key-value pair will have the following format:

```
return:ok|NumberOfRulesApplied:<number of applied
rules>|RulesApplied:<rule 1 id> <rule1 name>, <rule2 id> <rule2
name>, ...
```

Here is an example of what the result might look like:

```
AttributeUserData [List, size (unpacked)=168] =
'ESP_Result' [str] =
"return:ok|NumberOfRulesApplied:12|RulesApplied:McrSlt1GlbClsf1
McrSlt1GlbClassification1, McrSlt1GlbClsf2
McrSlt1GlbClassification2"
```

8. Verification is done to check if a business process was assigned by a classification rule.
9. If a business process was assigned, then the interaction is placed in the `iWD_Captured` queue.

10. If a business process was not assigned in [Step 8](#), then the interaction is placed in the `iWD_ErrorHeld` queue.
11. If rules could not be applied, the reason is provided.
12. The reason is analyzed, and a decision is made whether another attempt should be made to apply the rules.
13. The last error is attached to user data as a key-value pair with the key `ESP_Error`. The following is an example of what the result could look like:
`AttributeUserData [List, size (unpacked)=143] = 'ESP_Error' [str] = "605 Department 'McrSlit1_D1' has expired on 2010-08-15T00:00:00"`
14. The last error is analyzed. If error is between 604 and 607, that means the Department/Process is not active yet.
15. If the department or process is not active yet, the interaction is placed in the `iWD_Rejected` queue.
16. For all other errors, the interaction is placed in the `iWD_ErrorHeld` queue.
17. A check is done to see if the maximum number of attempts (see [Step 1](#)) has been reached.
18. If the maximum number of attempts has been reached, the most recent error is attached to the interaction and the interaction is placed in the `iWD_ErrorHeld` queue.
19. If the maximum number of attempts has not been reached, the counter is increased by 1.
20. The interaction is delayed for the amount of milliseconds specified in `_delay` (see [Step 1](#)) before another attempt is made to apply the rules.

Prioritization Strategy

The purpose of this strategy is to invoke the corresponding prioritization rules, analyze the result of the rules application and place the interaction into the appropriate queue, depending on the result.

This strategy processes interactions from the following queues:

- `iWD_Captured`

Interactions have to satisfy following conditions (see [Figure 79](#)):

- Active interactions only, (interactions which do not have the property `IWD_activationDateTime` set, or this property has a time stamp which is in the past.
- Interactions are taken in the order they were submitted.

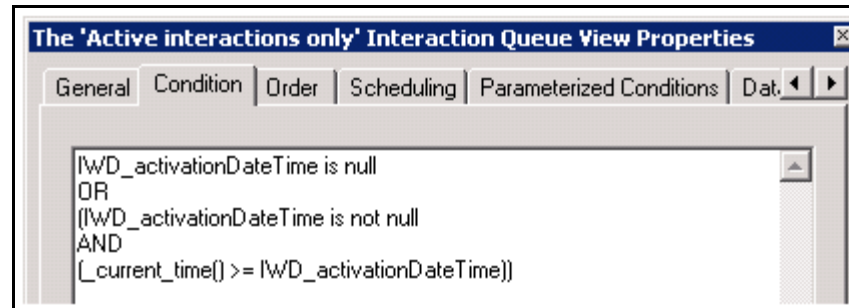


Figure 79: The 'Active interactions only' Interaction Queue View Properties

- iWD_Queued

Interactions have to satisfy the following conditions (see [Figure 80](#)):

- Interactions that are subject for immediate reprioritization (interactions that have the property `IWD_reprioritizeDateTime` set, and this property has a time stamp which is in the past)
- Interactions are taken in order of `IWD_reprioritizationDateTime` (oldest first).

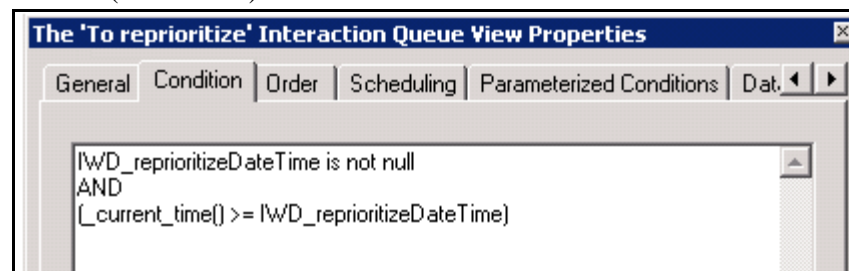


Figure 80: The 'To reprioritize' Interaction Queue View Properties

[Figure 81](#) shows the Prioritization strategy.

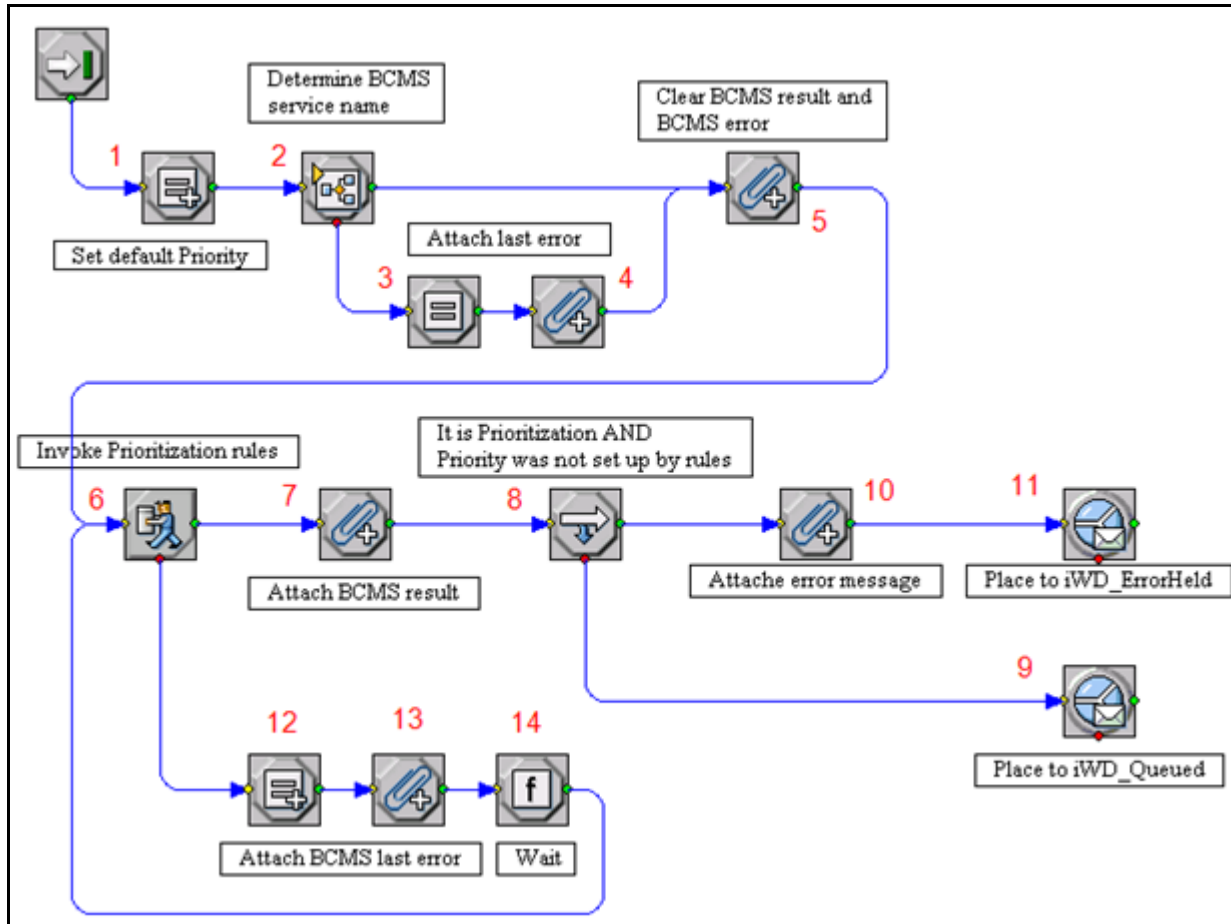


Figure 81: Prioritization Strategy

Summary of Flow

Note: The IRD objects described in this section are keyed to the numbers in Figure 81.

1. Variables are initialized:
 - `source_queue` is the queue from which the interactions came. It will be used to determine if the prioritization service is being requested for initial prioritization or reprioritization.
 - `BCMS_error_timeout_ms` specifies the delay (in milliseconds) between attempts to invoke rules.
 - `_default_priority` specifies the priority which will be assigned if a priority is not specified by the customer (as part of the task capture) or by rules.

2. The subroutine that determines the correct ESP server name is invoked. This subroutine also sets up cases when there is reason to retry to invoke the ESP server.
3. If the subroutine fails an error is extracted.
4. This error is attached to user data as a key-value pair with the key `IWD_ESP_Determination_Error`.
5. The User Data key-value pairs `ESP_Result` and `ESP_Error` are set to an empty string.
6. A request to apply prioritization rules to the interaction is sent.
7. If the ESP server reports that the operation was completed successfully, the result is attached to user data as a key-value pair with the key `ESP_Result`. This key-value pair will have following format:

```
"return:ok|NumberOfRulesApplied:<number of applied
rules>|RulesApplied:<rule 1 id> <rule1 name>, <rule2 id> <rule2
name>, "
```

The following is an example of what the result could look like:

```
AttributeUserData [List, size (unpacked)=168] =
'ESP_Result' [str] =
"return:ok|NumberOfRulesApplied:2|RulesApplied:McrSlT1GlbPrior1
McrSlT1GlbPrioritization1, McrSlT1GlbClsf2
McrSlT1GlbPrioritization2"
```
8. A check is made as to whether it is the initial prioritization, and whether the priority was set.
9. If the request to apply prioritization rules is for reprioritization, or if the priority was set to a value (other than the default value set by the BCMS service, which is 1), then the interaction is placed in the `iWD_Queued` queue.
10. If the request to apply prioritization rules is for initial prioritization, and priority was not set in rules, then the following error message is attached: `Priority is not set up by rules`.
11. The interaction is placed in the `iWD_ErrorHeld` queue.
12. If rules could not be applied, the reason is provided.
13. The most recent error is attached to user data as a key-value pair with the key `ESP_Error`. The following is an example of what the result will look like if prioritization rules were not configured:

```
attr_changed_prop [List, size (unpacked)=65] = 'ESP_Result' [str] =
"return:ok|NumberOfRulesApplied:0|RulesApplied:"
```
14. The interaction is delayed for the amount of milliseconds specified in `_BCMS_error_timeout_ms` (see [Step 1](#)) before another attempt is made to apply the rules.

Distribution Strategy

This simple strategy routes interactions to a requested Agent, requested Agent Group, requested Skill, or to the default iWD Agent Group.

This strategy processes interactions from the following queues:

- iWD_Queued

Interactions have to satisfy the following conditions (see [Figure 82](#)):

- Interactions that are not subject for immediate reprioritization (interactions that do not have the property `IWD_reprioritizeDateTime` set, or that have this property set to a time stamp that is in the future).
- Interactions are taken in order of priority (highest priority first)

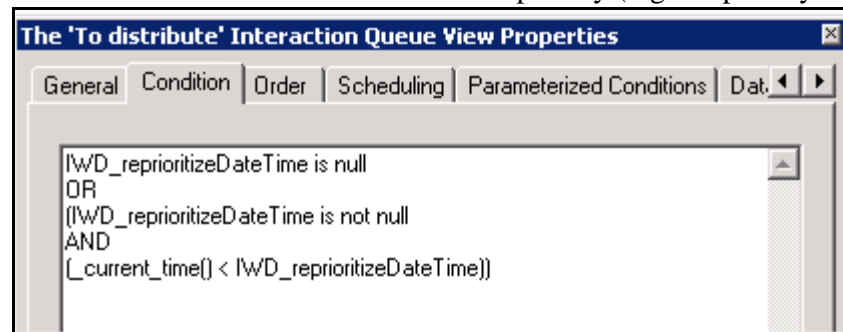


Figure 82: The 'To distribute' Interaction Queue View Properties

[Figure 83](#) shows the Distribution strategy.

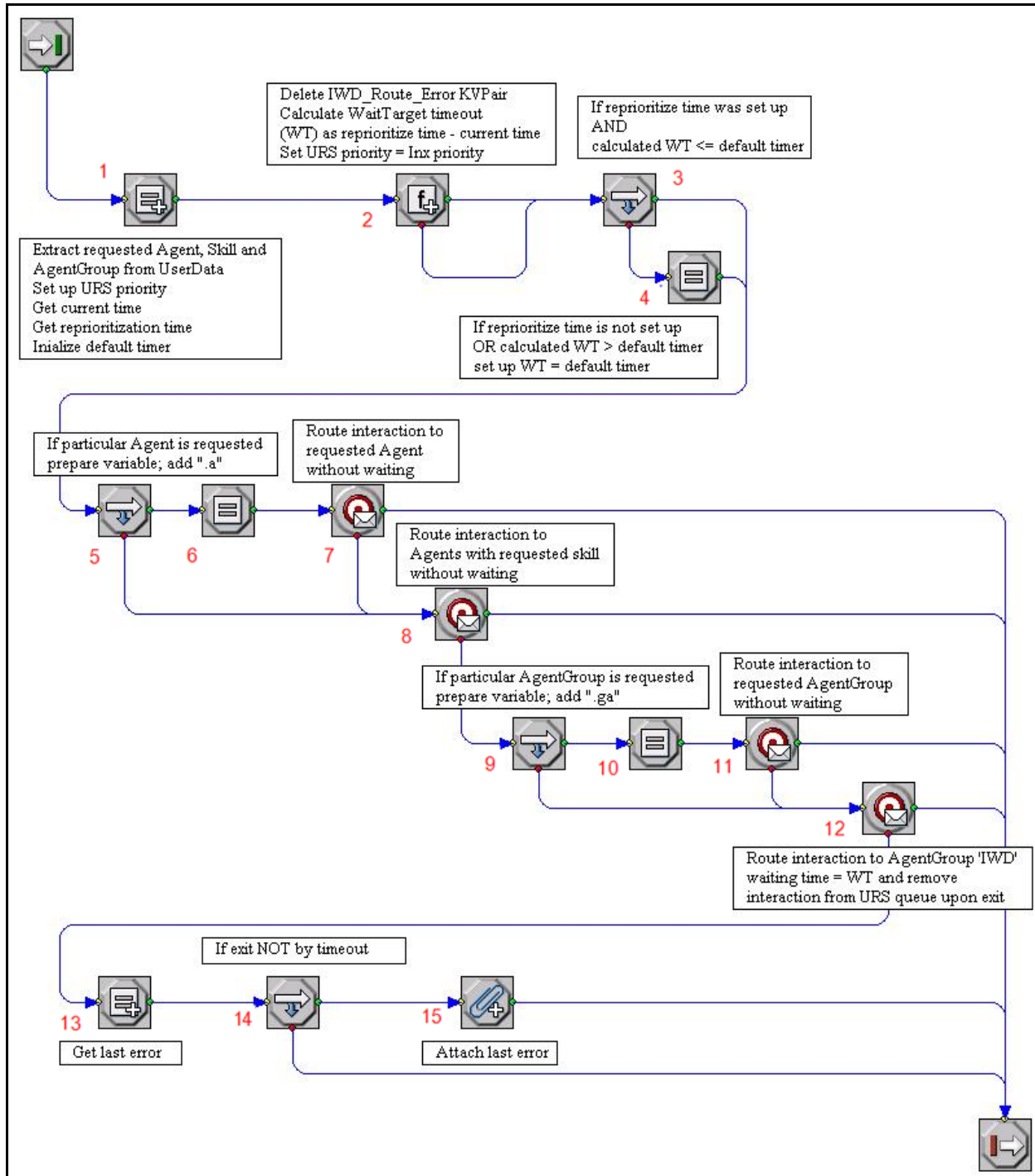


Figure 83: Distribution Strategy

Summary of Flow

Note: The IRD objects described in this section are keyed to the numbers in [Figure 83](#).

1. Extract information about requested agent and skill and initialize internal variables. See [Figure 84](#).

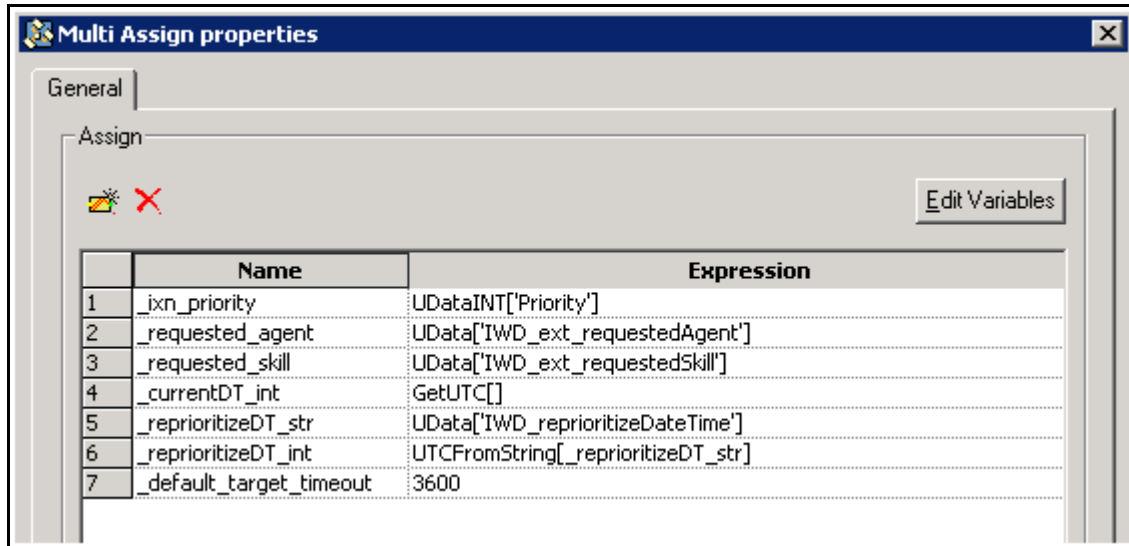


Figure 84: Requested Agent and Skill

2. A calculation is done to determine the timeout—how long the interaction should wait for a target to become available.
3. If the reprioritize time was set up and the calculated timeout is less than or equal to the default timeout (1 hour, see [Step 1](#)), then the timeout remains as it is.
4. If the reprioritize time was not set, or the calculated timeout is greater than the default timeout, then the waiting timeout is set to the default (1 hour).
5. Analysis is done to determine whether an agent was requested.
6. If an agent was requested, the URS variable is prepared (.a is added).

7. Try to route the interaction to the requested agent without waiting (see [Figure 85](#)).

Route Interaction properties

Interaction Queue Target Selection

Statistics

☐ Min ☐ Max Name

Targets

☒ Clear Target Timeout 0 Sec

	Type	Name	StatServer
1	Variable	_requested_agent	

Figure 85: Route to Agent

8. Try to route the interaction to an agent with the requested skill without waiting (see [Figure 86](#)).

Route Interaction properties

Interaction Queue Target Selection

Statistics

☒ Min ☐ Max Name StatAgentLoading

Targets

☐ Clear Target Timeout 0 Sec

	Type	Name	StatServer
1	Skill	_requested_skill >	

Figure 86: Route to Skill

9. Analysis is done to determine whether an Agent Group was requested.
10. If an Agent Group was requested, the URS variable is prepared (.ga is added).

11. Try to route the interaction to the requested Agent Group without waiting (see [Figure 87](#)).

	Type	Name	StatServer
1	Variable	_requested_agent_group	

Figure 87: Route to Requested Agent Group

12. Try to route the interaction to the iWD agent group with a wait time of 60 seconds (see [Figure 88](#)).

	Type	Name	StatServer
1	Agent Group	IWD	

Figure 88: Route to Agent Group

13. Get the last error.
14. Verification is done as to why the target was not found.
15. An error code is attached in case of any error other than a timeout.

If more than one target is available, URS uses the StatAgentsLoading statistic to select the Agent who has the minimum load (this applies to routing to Skills and routing to Groups only; routing to a requested Agent does not use statistics). For more information about this statistic, see the *Universal Routing 8.0 Reference Manual*.

The Route Interaction object also has an **Interaction Queue** tab. See [Figure 89](#). (This applies to all three Route Interaction objects in this strategy.)

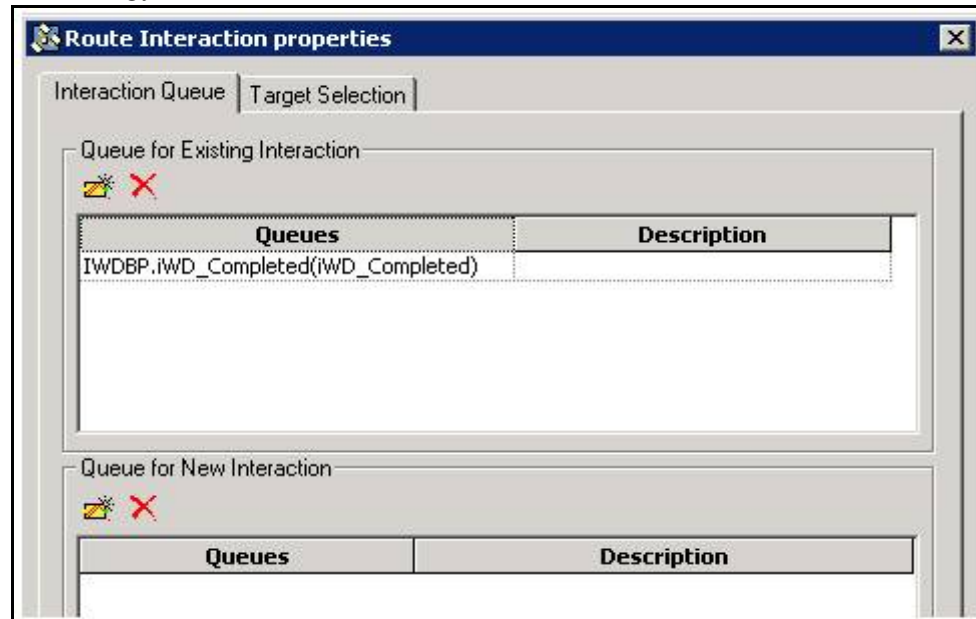


Figure 89: Route Interaction Properties—Interaction Queue

The optional **Interaction Queue** tab enables you to specify two types of queues:

- Queues for existing interactions (the queue in which the interaction should be placed after the agent is done working with it).
- Queues for new interactions (the queue in which new interactions created by the agent should be placed).

A **Description** (optional) appears as a hint on the agent desktop as to where to place the interaction.

Archive Strategy

The purpose of this strategy is to invoke the corresponding archiving rules, analyze the result of the rules application, and process the interaction accordingly. There are three archive destinations: delete the interaction, reschedule archiving for some time in the future, and archive the interaction to another Interaction Server database. A key-value pair in user data with the key `IWD_expirationDateTime` contains information about when an interaction has to be archived.

This strategy processes interactions from following queues:

- `iWD_Completed`
- `iWD_Canceled`
- `iWD_Rejected`

Interactions have to satisfy the following conditions (see [Figure 90](#)):

- Interactions that are not subject for archiving (interactions that do not have the property `IWD_expirationDateTime` set, or this property has a time stamp which is in the past).
- Interactions are taken in the order they were submitted.

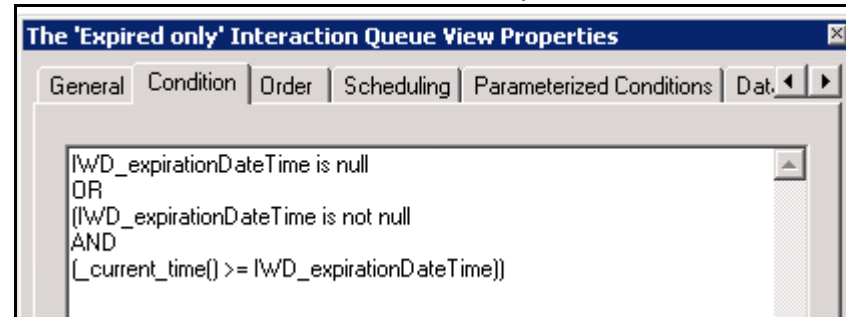


Figure 90: The 'Expired only' Interaction Queue View Properties

Figure 91 shows the Archive strategy.

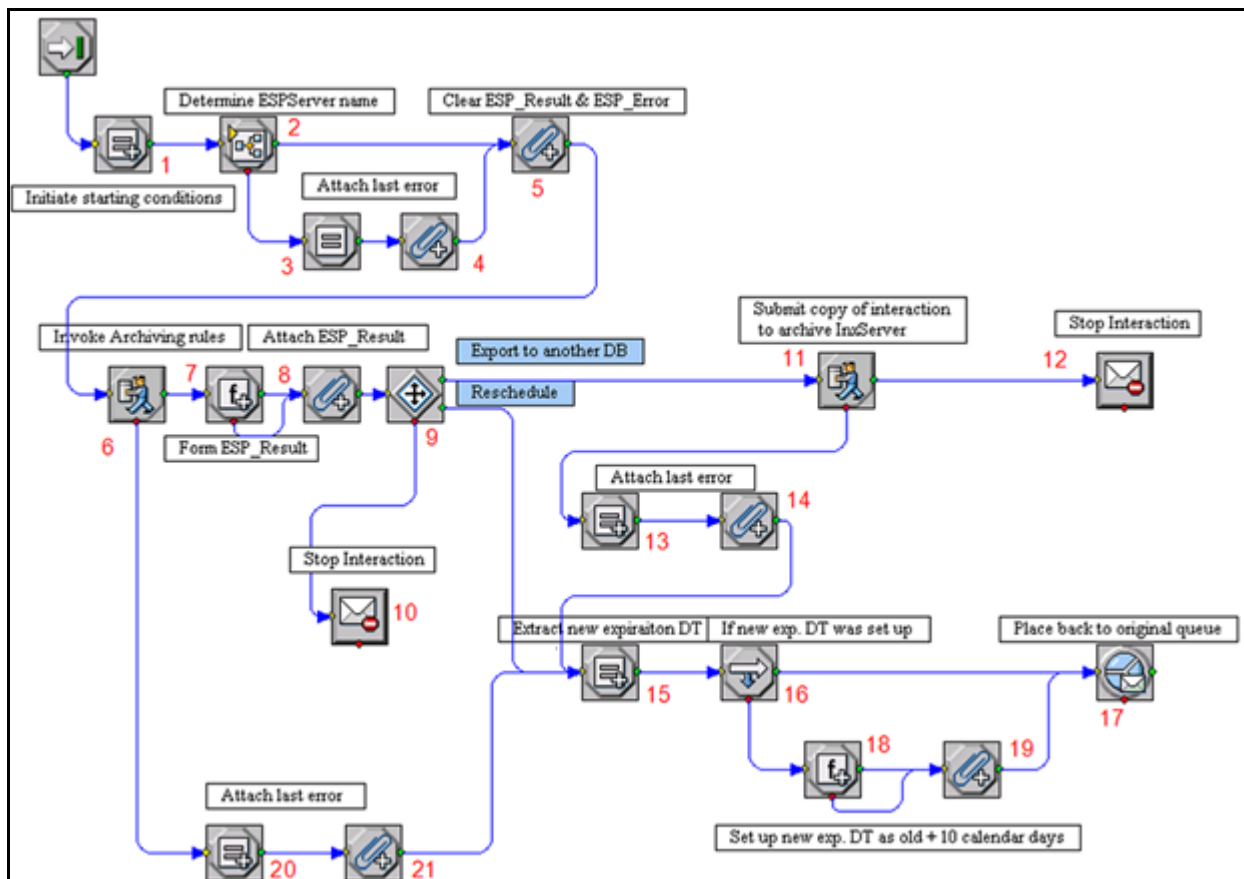


Figure 91: Archive Strategy

Summary of Flow

Note: The IRD objects described in this section are keyed to the numbers in [Figure 91](#).

1. Variables are initialized (see [Figure 92](#)):
 - `_default_reschedule` is the default interval (in seconds) for archive rescheduling. The default value is 864000 (10 days). The default value will be used if archiving rules were not configured, a new archiving time was not configured, or in case of error.
 - `_old_expiration_dt_str` is a string that is taken from the interaction before any archive rules are invoked.
 - `_old_expiration_dt_int` is an integer that is taken from `_old_expiration_dt_str`.

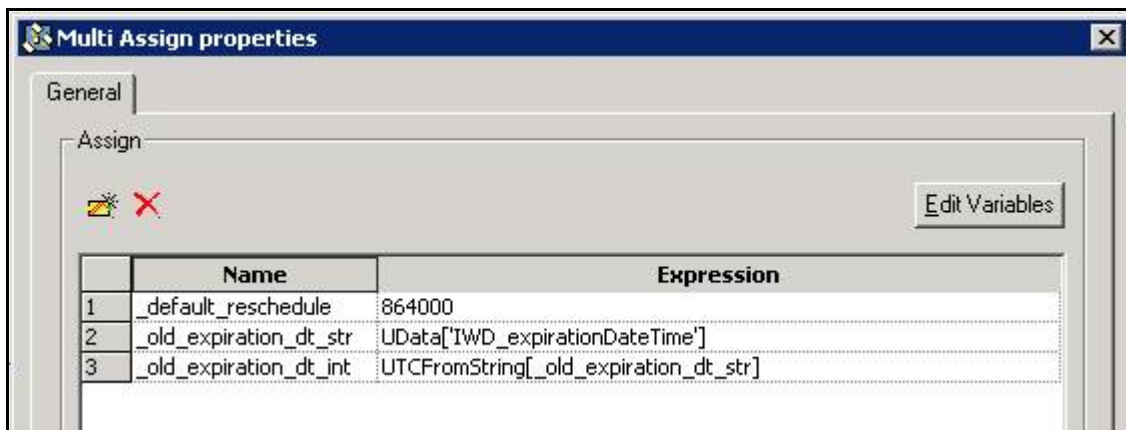


Figure 92: Archive Strategy—Initialize Variables

2. The subroutine that determines the correct ESP server name is invoked. This subroutine also sets up cases when there is reason to retry to invoke the ESP server. A retry mechanism is not included. See “Classification Strategy” on [page 256](#) for an example.
3. If the subroutine fails an error is extracted.
4. This error is attached to user data as a key-value pair with the key `IWD_ESP_Determination_Error`.
5. The User Data key-value pairs `ESP_Result` and `ESP_Error` are set to an empty string.
6. A request to apply archiving rules to the interaction is sent.
7. The ESP result is formed in the following format:


```
"return:ok| NumberOfRulesApplied:<number of applied
rules>|RulesApplied:<rule1id> <rule1name>, <rule2id> <rule2name>,
..."
```
8. The result is attached to User Data as a key-value pair with the key `ESP_Result`.

9. The result of the rules application is analyzed. See [Figure 93](#).

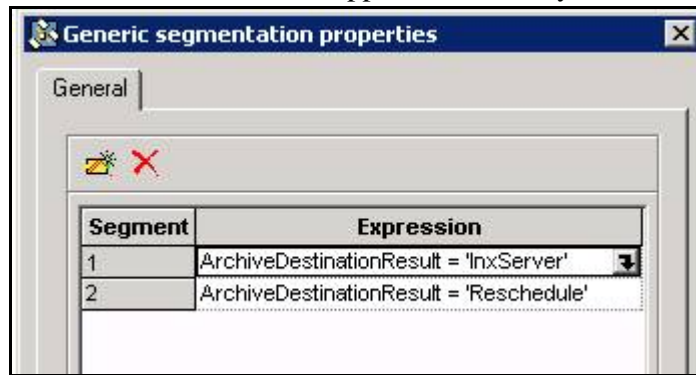


Figure 93: Result of Rules Application

10. Any archiving destination except those listed in [Figure 93](#) will cause processing of the interaction to stop. The interaction will also be deleted from the Interaction Server runtime database.
11. An exact copy of the interaction will be submitted to the Interaction Server that is specified in the Application Name field of the External Service block. See [Figure 94](#).

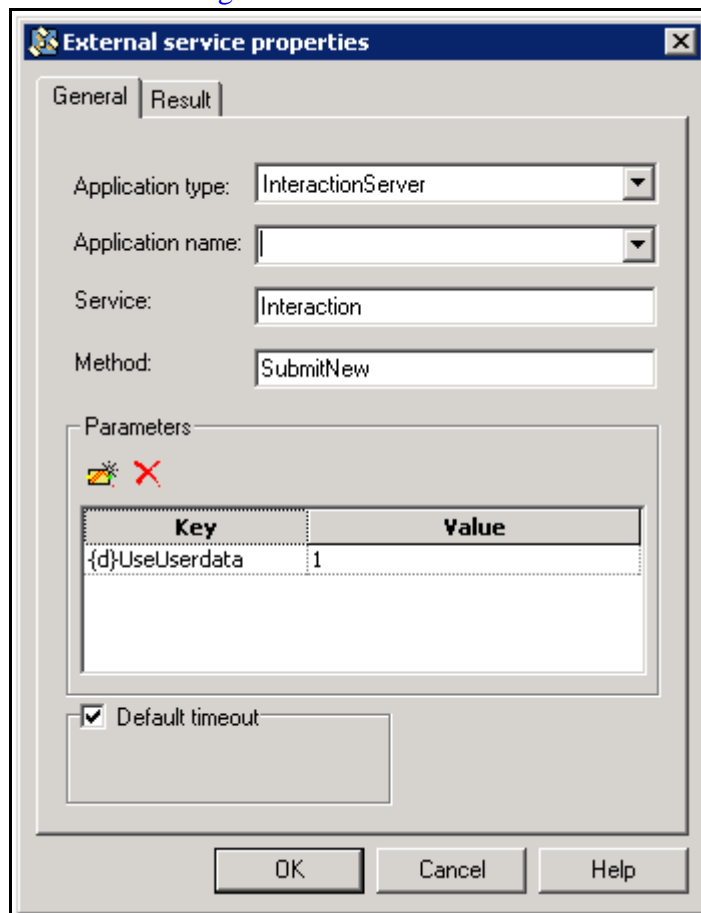


Figure 94: Specify Interaction Server

12. If the submission is a success, processing will be stopped on the interaction and the interaction will be deleted from the Interaction Server runtime database.
13. If the submission failed, the last error is retrieved.
14. The most recent error is attached to User Data as a key-value pair with the key ESP_Error.
15. If a new expiration timestamp exists, it is extracted.
16. Analysis is done to determine if a new expiration timestamp was set up.
17. If the new expiration timestamp is different from the old one, it means that archiving rules were configured and applied correctly. Therefore, the interaction is returned to the original queue for future archiving.
18. If there is no new expiration timestamp, a default timestamp is created. See [Figure 95](#).

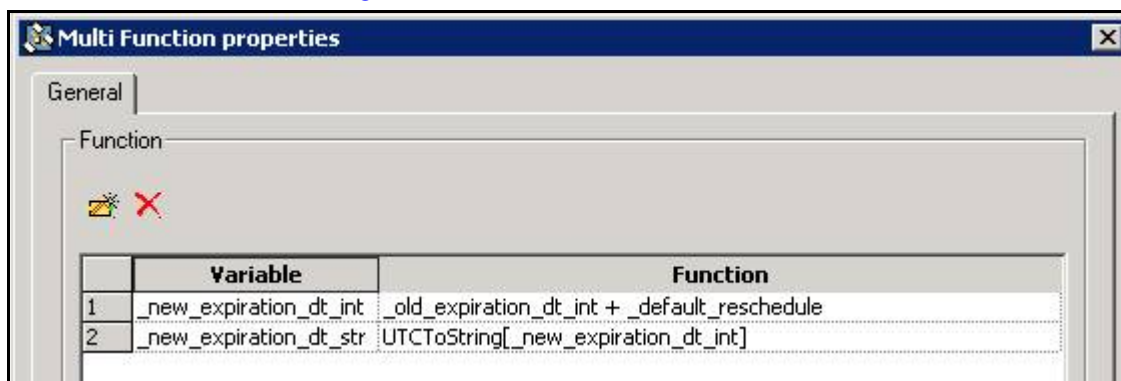


Figure 95: Default Timestamp is Created

19. The new expiration timestamp is attached to user data as a key-value pair with the key IWD_expirationDateTime.
20. The reason why the operation failed is received.
21. The most recent error is attached to user data as a key-value pair with the key ESP_Error. The following is an example of what the result could look like:

```
AttributeUserData [List, size (unpacked)=130] = 'ESP_Error' [str] =
"608 No archive destination set by archiving rules"
```

Modifying the iWD Business Process

For most environments, the only modification that will need to be made to the iWD Business Process is to the Distribution strategy (see [page 263](#)). The recommended approach to do this is to add a new strategy into the iWD Business Process, and replace the connection from iWD_Queued/ALL view to the Distribution routing strategy with a connection from iWD_Queued to your own

routing strategy where distribution logic is described. Then, link your new distribution strategy to the out of the box `iWD_Completed` queue.

By modifying the business process in this way, rather than simply updating the provided Distribution strategy, you can easily import any new versions of the iWD Business Process that might be available in the future (the links will have to be reestablished to your own distribution strategy).



Appendix

C

Adapting the iWD Business Process for Standard Genesys Channels

This Appendix describes how to adapt the iWD Business Process (IWDBP) to work with standard Genesys non-voice channels, such as e-mail, chat, SMS, Gplus Adapters, and custom integrations built with Open Media.

This Appendix contains the following sections:

- [Adapting IWDBP to Serve E-mail Interactions, page 275](#)
- [Examples, page 281](#)

Adapting IWDBP to Serve E-mail Interactions

This section explains what should be done to adapt the iWD business process to serve e-mail interactions.

Adding Required Properties to Interactions

In order to keep Data Mart functionality intact and to make Genesys standard channel interactions visible in iWD Manager, some key-value pairs need to be added to the user data of these interactions. The interactions should only be placed into the input queue for the default iWD business process *after* these key-value pairs have been added.

The key-value pairs are:

- IWD_tenantId
- IWD_solutionId

- IWD_capturePointId

To make the process easier, the iWD Setup Utility includes an additional business process, Standard Genesys to IWD adapter (see [Figure 96](#)). This business process attaches the required key-value pairs to an interaction and places it into the input queue of the default business process IWDBP.

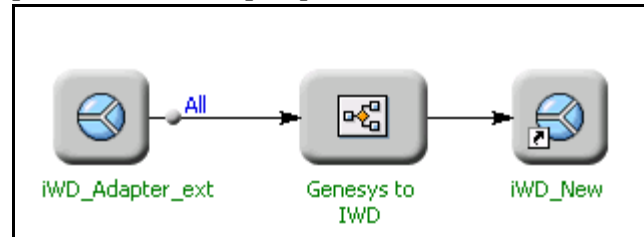


Figure 96: Standard Genesys to iWD Adapter Business Process

[Figure 97](#) shows the Genesys to IWD strategy.

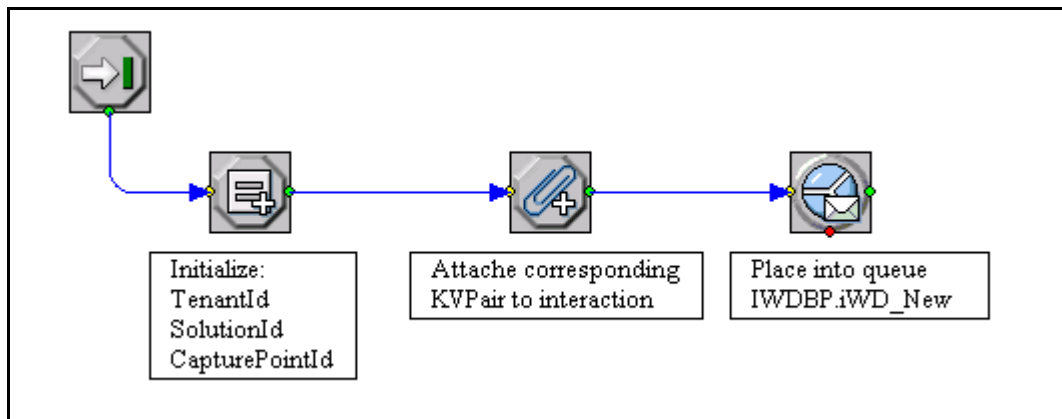


Figure 97: Genesys to iWD Strategy

In the Multi Assign object, you have to initialize all variables, as shown in [Figure 98](#).

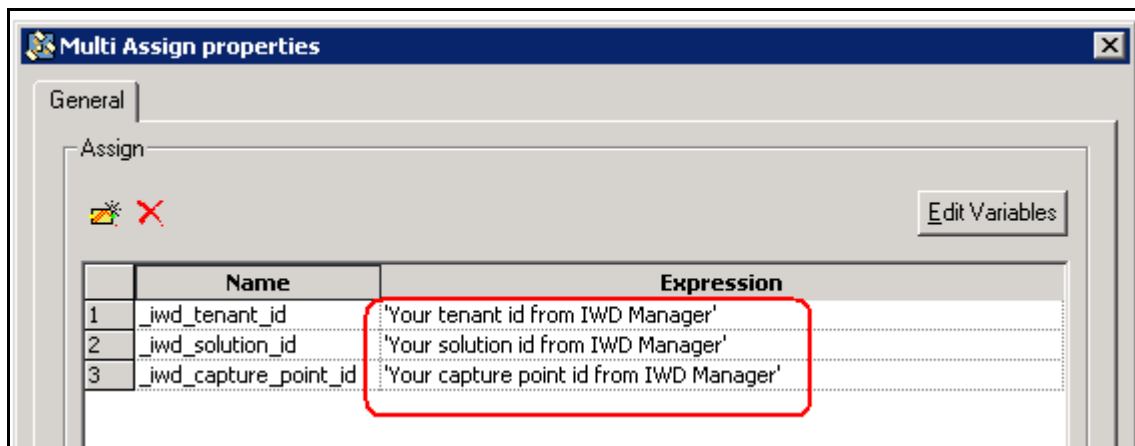


Figure 98: Assign Variables

The IDs are taken from iWD Manager. [Figure 99](#) and [Figure 100](#) show where to find your Tenant ID and Solution ID, respectively.

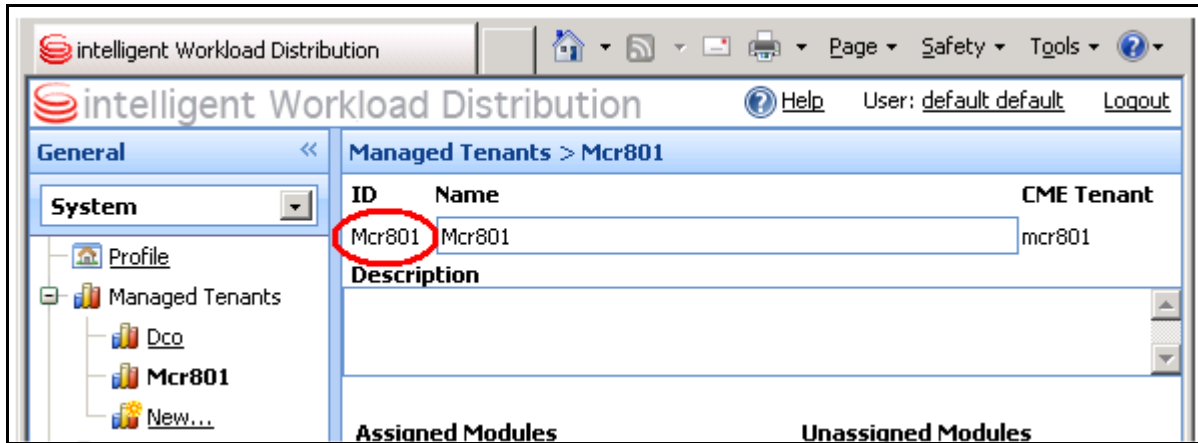


Figure 99: Tenant ID

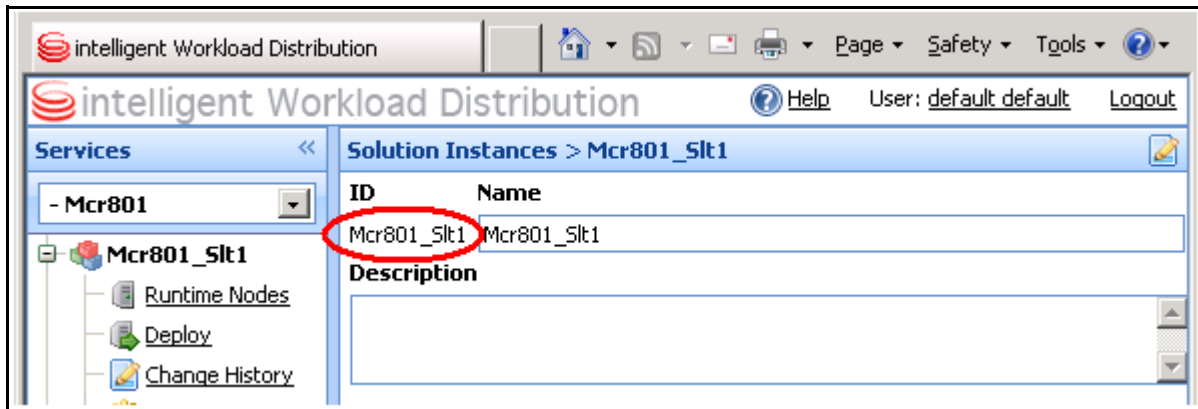


Figure 100: Solution ID

In order to get an ID for a capture point you have to create a “dummy” capture point. This will represent a Genesys standard server (in our example, E-mail server). Use the Webservice Capture Point template to create a simple “dummy” capture point. The ID is the only property that is needed. It does not matter what you enter for the rest of the properties. [Figure 101](#) shows the iWD Capture Point ID.

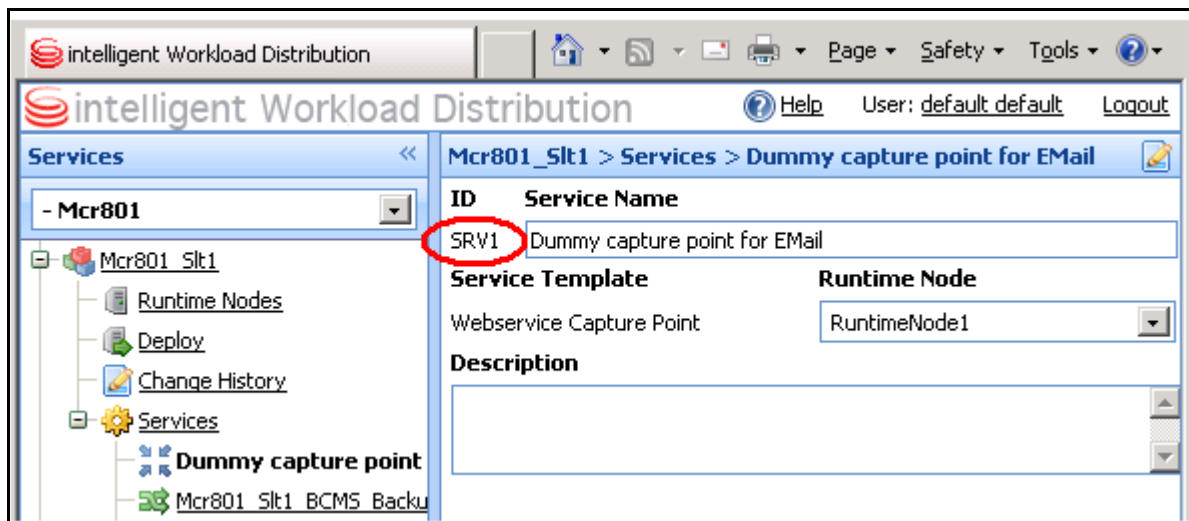


Figure 101: iWD Capture Point ID

Adding E-mail Server

E-mail Server needs to be added to the Standard Genesys to IWD Adapter business process. You can add E-Mail Server in two ways:

1. In Configuration Manager or Genesys Administrator, you can update the E-Mail Server application options to specify `iWD_Adapter_ext` as an output queue. Add a section called `endpoint:[YourTenantDBID]`. In this section, add a new option `default`. Set the value of `default` to `iWD_Adapter_ext`, as shown in Figure 102. Next, refresh in IRD and E-mail Server will be added to the business process with the `iWD_Adapter_ext` queue.

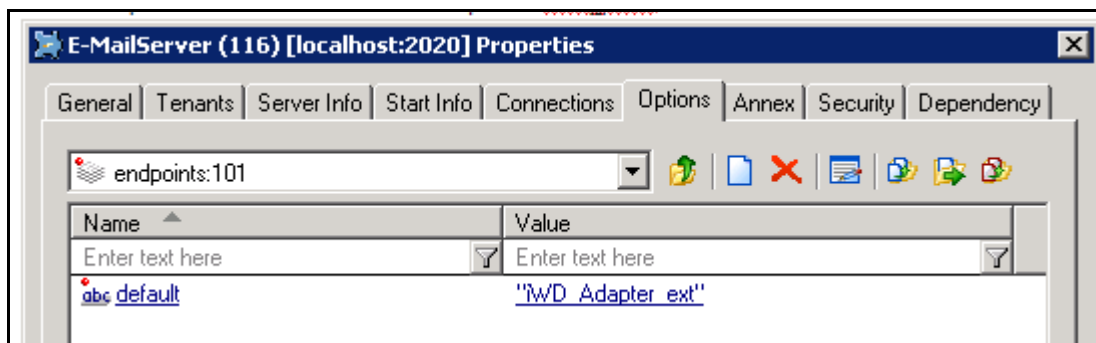


Figure 102: Updating E-mail Server for Standard Genesys to IWD Adapter Business Process

2. The second way to add E-mail Server is to do so explicitly in IRD. Add E-mail server from Media Servers to the Standard Genesys to IWD Adapter business process and make a connection to the `iWD_Adapter_ext` queue. In this method, IRD will update the corresponding option and section in the E-Mail Server application.

Figure 103 shows how the Standard Genesys to IWD Adapter business process will look after these modifications.

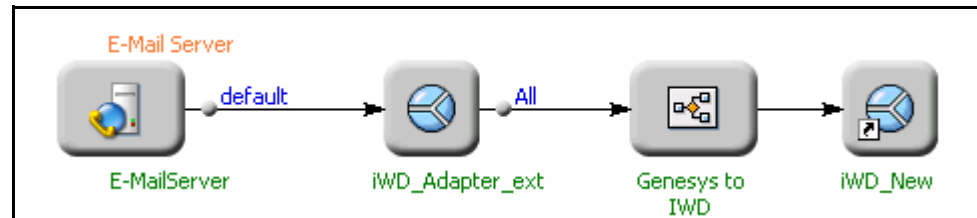


Figure 103: Standard Genesys to iWD Adapter Business Process After Modifications

Modify the Distribution Strategy

In IRD, open the IWDBP business process. Open the Distribution strategy. Since there are business actions Request Agent and Request Skill, the default business process has to take this into consideration. This is why there are three objects of type Route Interaction.

If you have only one Stat Server listed in the Connections tab of Interaction Server, you can skip this step. If not, for the object Route interaction to AgentGroup IWD in the Target Selection tab, change the Genesys Stat Server application name and target according to your configuration. Also you can change how long URS has to wait for the next available agent (by default it is set to 60 seconds). See Figure 104.

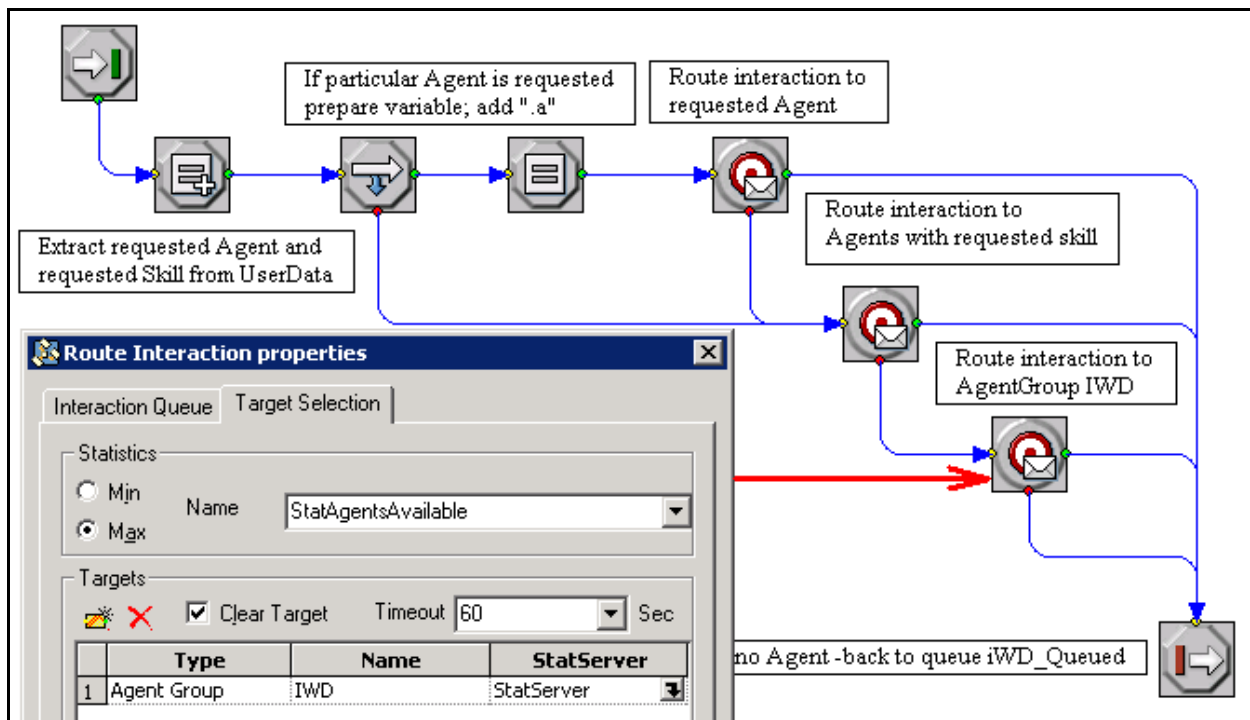


Figure 104: Update Route Interaction to Agent Group IWD

You might want to check all of the provided `Route Interaction` objects (by double-clicking on them) to see if they satisfy your business logic. Pay attention only to what statistics are used for `Skill` and `AgentGroup` routing.

Allowing Agents to Send Replies to Inbound E-mails

There are two ways to allow agents to send replies to inbound e-mails: create a new queue and a new strategy in the IWDBP, or use a business process that already exists. In this example an existing business process will be used to illustrate how other business processes can be used from the iWD business process.

First, choose a business process to handle agent's replies and outbound e-mails. Second, specify the queue into which the agent's reply will be placed. Third, specify the business process and queue for single outbound e-mails from agents.

In this example the `ABC Simple` business process will process agent's replies (with the `Outbound` queue). Also, agents will be given the ability to place interactions into the `iWD_Completed` queue. All of this will be done in the `Distribution` strategy. The `Route Interaction` properties in the strategy must be as shown in [Figure 105](#).

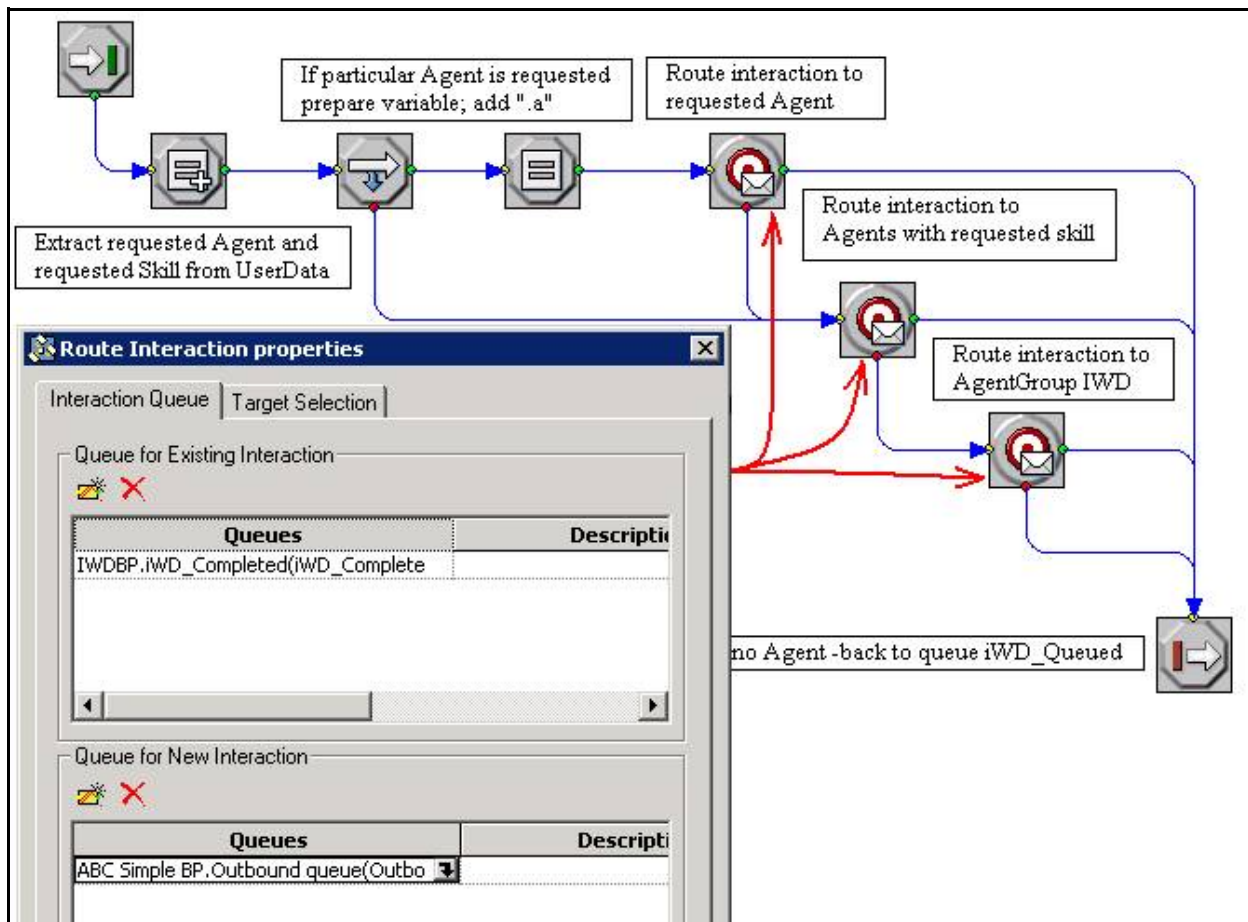


Figure 105:Route Interaction Object for Distribution Strategy

Examples

In these examples we assume that the default iWD Business Process (IWDBP) provides all necessary steps for e-mail processing—namely classification, prioritization, and distribution.

The purpose of these examples is to show what needs to be done in order to use IWDBP and standard iWD functionality (such as classification and prioritization rules) for e-mail processing.

Note: The following examples are presented as guidelines. Some of the strategies and objects in the business processes might not be exactly as shown in the following examples.

Requirements:

- A Genesys E-mail solution is installed
- An iWD solution is installed

Assumptions:

Only one iWD solution will be served by iWDBP. The default iWD business process will process interactions with any media type (the interaction will pass through the business process and be delivered to an agent), but business rules created in these examples will be applicable to e-mails only. We have only one Agent Group to which the interactions will be assigned.

For all examples, the main flow of iWDBP is as shown in [Figure 106](#).

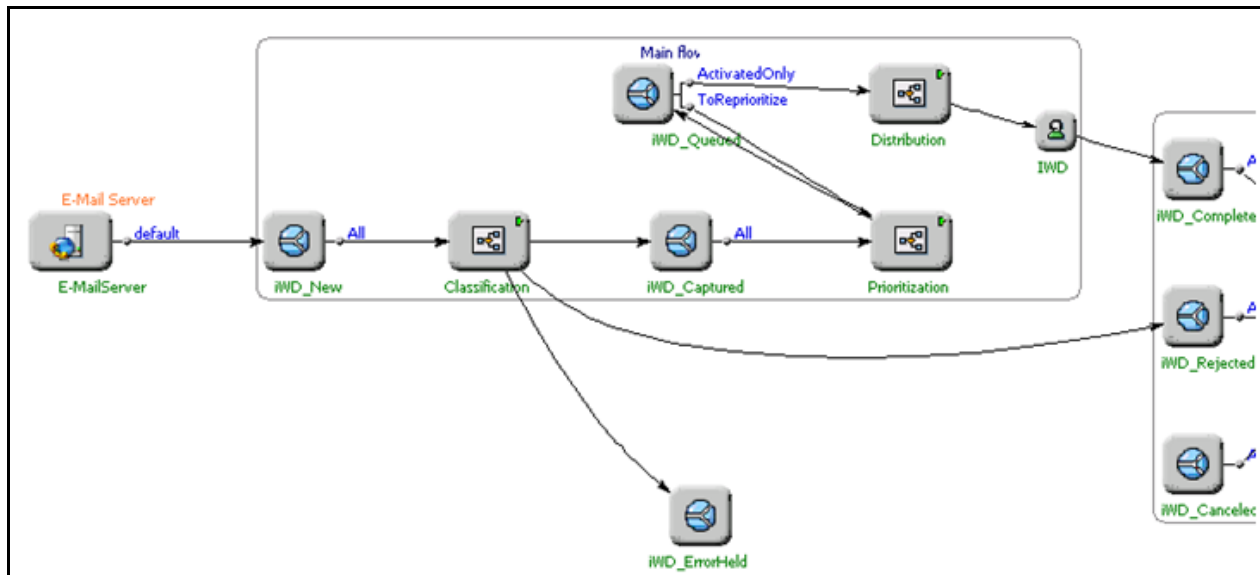


Figure 106: Main Flow of iWDBP

You can add E-Mail Server in two ways:

1. In Configuration Manager or Genesys Administrator, you can update the E-Mail Server application options to specify iWD_New as an output queue. Add a section called endpoint: [YourTenantDBID]. In this section, add a new option default. Set the value of default to iWD_New, as shown in [Figure 106](#). Next, refresh in IRD and E-mail Server will be added to the business process with the iWD_New queue.

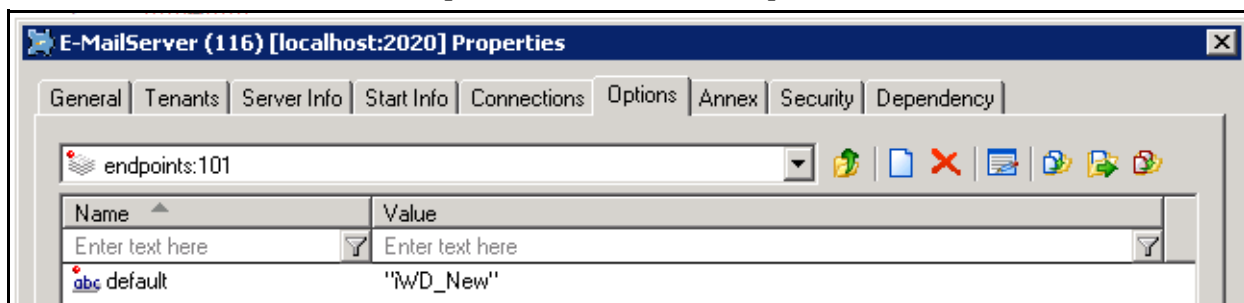


Figure 107:Update E-Mail Server Options

2. The second way to add E-mail Server is to do so explicitly in IRD. Add E-mail server from Media Servers to your business process and make a connection to the iWD_New queue. In this method, IRD will update the corresponding option and section in the E-Mail Server application.

Example 1

This is a simple example of how business rules can be used. In this example, the default iWD business process will be used for processing Genesys e-mails.

Use Case

In this example, the following scenario/use case is used:

- For all interactions with `MediaType = email`, the property `ServiceType` will be set to `ChangeAddress`.
- The property `priority` will be set to `100` for all e-mail interactions.
- Interactions of any `MediaType` should be delivered to the Agent Group `IWD` (interactions with the highest priority have to be delivered first).
- E-mail interactions have to be reprioritized every 2 hours.
- After each reprioritization the priority have to be increased by 5.

Preparation of Components**Genesys
Configuration**

To prepare the Genesys configuration:

- Add Agents into the IWD Agent Group.
- Set up a connection between Interaction Server and the External Service Provider service. According to IWD BP, Interaction Server will send interactions to an iWD External Service Provider service (ESPService) to apply business rules. In order to do that, we have to “explain” to Interaction Server where to find the ESPService. This is done by adding an Application that represent the ESPService, to the Connections of the Interaction Server application.
- Set the proper outbound queue for E-mail Server. Interactions that are submitted by E-mail Server have to reach the iWD business process in some way. In order to do that we need to change the outbound queue for the E-mail Server application to `iWD_New` in the endpoints section (refer to the first method of adding E-mail Server to the business process on [page 282](#) for details). Now your E-mail Server will submit interactions into the `iWD_New` queue, which is the entry point for the default iWD business process.

iWD Configuration To prepare the iWD configuration:

Note: It is recommended to give meaningful names for iWD services and objects. The following format could be useful:

`<iWDTenantName><iWDSolutionName><ServiceTypeServiceName>` or
`<iWDTenantName><iWDSolutionName><ParentObjectNameObjectName>`.

- Login into iWD Manger by using the default person account.
- Create a new iWD tenant. From the drop-down list choose the corresponding Genesys Tenant. It is recommended to give the new iWD tenant the same name as the Genesys tenant. In this example the iWD tenant name will be Mcr.
- Under iWD tenant Mcr create a new solution and name it McrSlt.
- Under iWD tenant Mcr create a new Role under Security Policy and give some permissions.
- Under the new solution create the following iWD configurations objects:
 - iWD Runtime Node. For Context URL use the directory name `iwd_node` provided during installation (by default it should be `http://localhost:8080/iwd_node/`). For the Application property, use the iWD Runtime Node application as configured in Configuration Manager or Genesys Administrator.
 - iWD Rules service
 - iWD Logging service
 - iWD Configuration Server Connector service
 - iWD Interaction Server Connector service
 - iWD Business Context Management service. For the Application property, use the iWD Business Context Management Service application as configured in Configuration Manager or Genesys Administrator.

IWDBP Preparation To prepare the iWD business process:

- In IRD open IWDBP. Open the Distribution strategy. Double-click on the RouteInteraction block. In the Target Selection tab, change the Genesys Stat Server application name and target according to your configuration. You can also change how long URS has to wait for an available agent. By default it is set to 60 seconds. See [Figure 108](#).

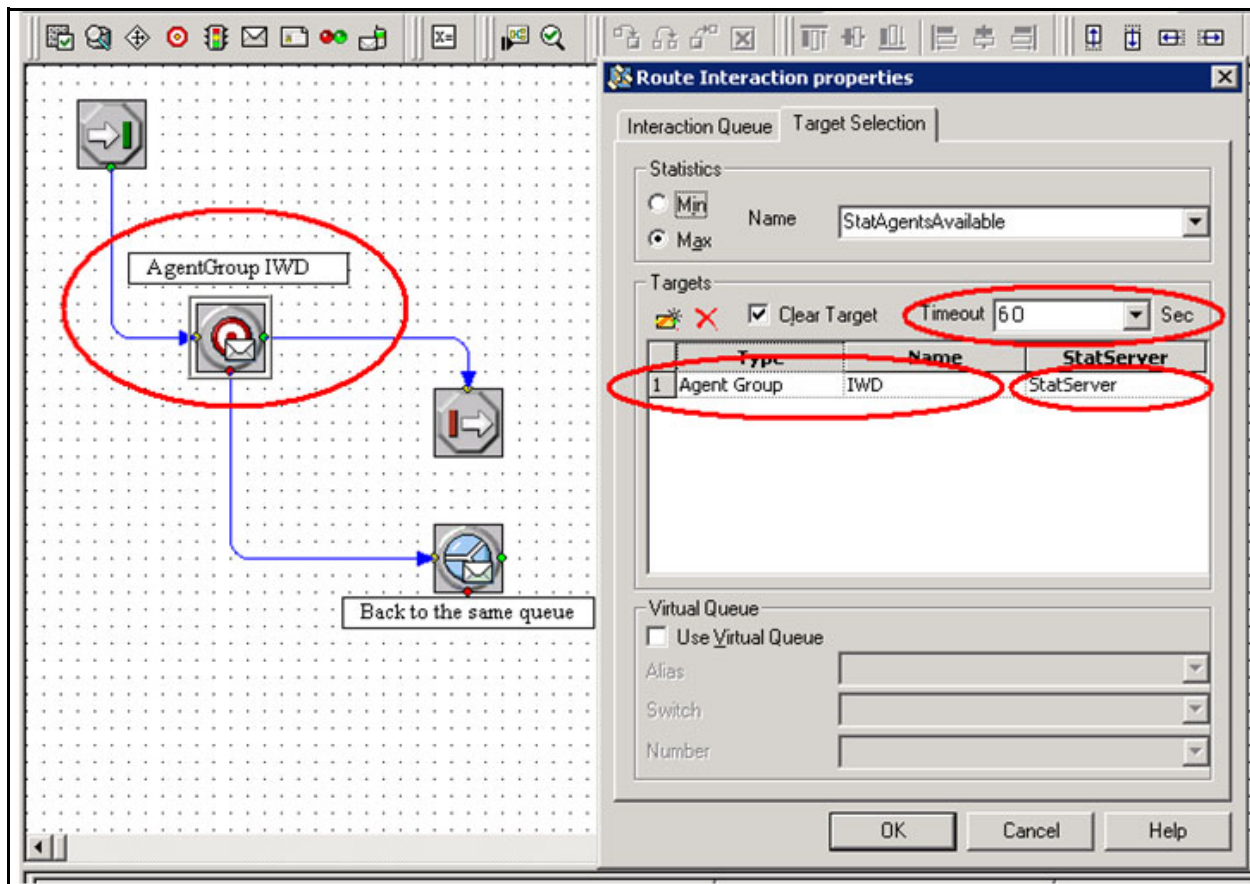


Figure 108:Preparing the iWD Business Process

Create Rules

Modify the Standard Rules Template

For simplicity, in this example conditions will be added to the Standard Rules Template, which is used for all iWD tenants.

1. Under Modules & Components choose the System tenant, expand Rule Templates and choose Standard Rules Template. Click the add icon (labeled as 1 on [Figure 109](#)).
2. In the Language Expression column, add a description for the condition (see [Table 61](#)).
3. In the Rule Language Mapping column enter the corresponding text (see [Table 61](#)).

4. Click Save.

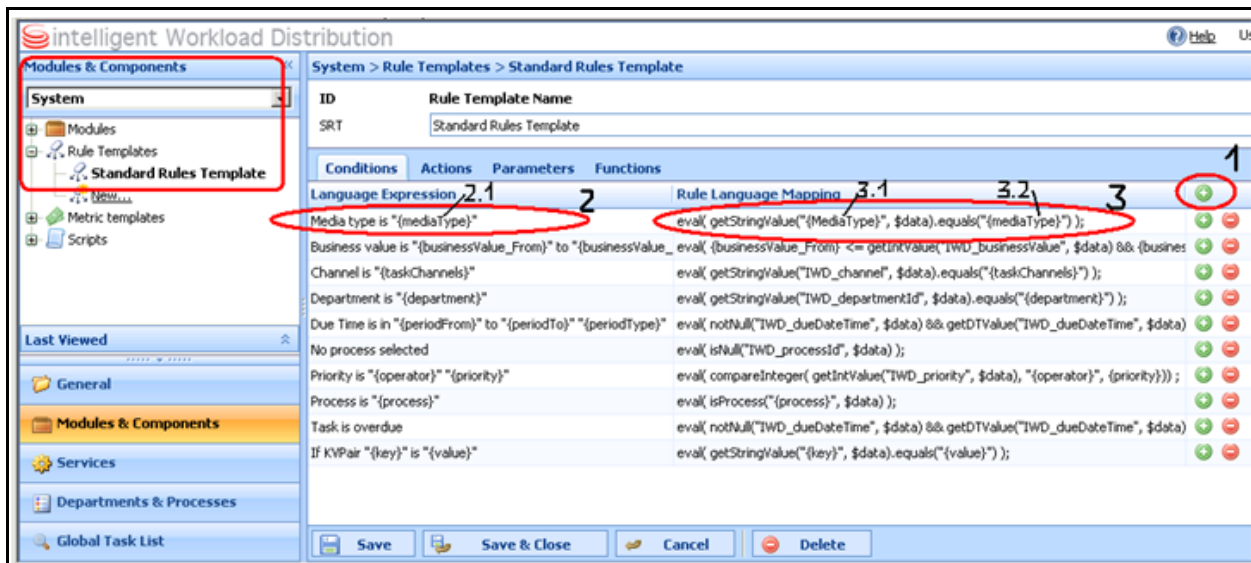


Figure 109: Add a Condition to the Standard Rules Template

Add more Conditions. Enter the text according to [Table 61](#).

Table 61: Conditions

Language Expression	Rule Language Mapping
If Reprioritization was not set up	eval(isNull("IWD_reprioritizeDateTime", \$data)); Note: The Standard Rules Template contains a standard rule condition called "Is first prioritization" that does the same thing as "If Reprioritization was not set up", but "If Reprioritization was not set up" is included in this example to give you another working example of how to achieve the same result in a business rule.
If Reprioritization was set up and in the past	eval(notNull("IWD_reprioritizeDateTime", \$data) && getDTValue("IWD_reprioritizeDateTime", \$data).before(getCurrentDT()));

Table 61: Conditions (Continued)

Language Expression	Rule Language Mapping
If KVPair "{k}" is "{v}"	eval(getStringValue("{k}", \$data).equals("{v}")); Note: The Standard Rules Template contains a standard rule condition called "String "{attribute}" equals "{stringValue}" that does the same thing as If KVPair "{k}" is "{v}". If KVPair "{k}" is "{v}" is included in this example to give you another working example of how to achieve the same result in business rules.
If KVPair "{k}" is not "{v}"	If KVPair "{k}" is not "{v}"eval(notNull("{k}", \$data) && !getStringValue("{k}", \$data).equals("{v}"));

To make this example as flexible as possible, add a universal action. This action will make it possible to add an key-value pair to user data. Click on the Action tab, and add a new action. In the Language Expression column enter:

Attach Custom KVPair "{key}" "{value}.

In the Rule Language Mapping column enter:

setStringValue("{key}", "{value}", \$data);

Note: The Standard Rules Template contains a standard rule action that does the same thing as Attach Custom KVPair: Set string "{attribute}" value "{stringValue}". Attach Custom KVPair is used here as an example of how to configure an action.

Create a Classification Rule

Create a classification rule. The numbers in the following steps are keyed to [Figure 110](#). For simplicity Global Rules are used in this example, but these classification rules could also be created at the Capture Point level, if a Capture Point ID was added to the interaction as described in “Adding Required Properties to Interactions” on [page 275](#).

1. Select Departments & Processes.
2. Select your iWD tenant.
3. Select Global Rules.
4. Select New Linear Rule.
5. Enter a name for the new rule.
6. Select classification as the Phase.
7. Click Add condition and select a condition.
8. The selected condition is added.
9. Select e-mail.

10. Click Add action.
11. Select Attach Custom KVPair.
12. Enter ServiceType and ChangeAddress.
13. Also add Set Priority 100.
14. For simplicity we will set up time for Reprioritization here.
15. Click Save.

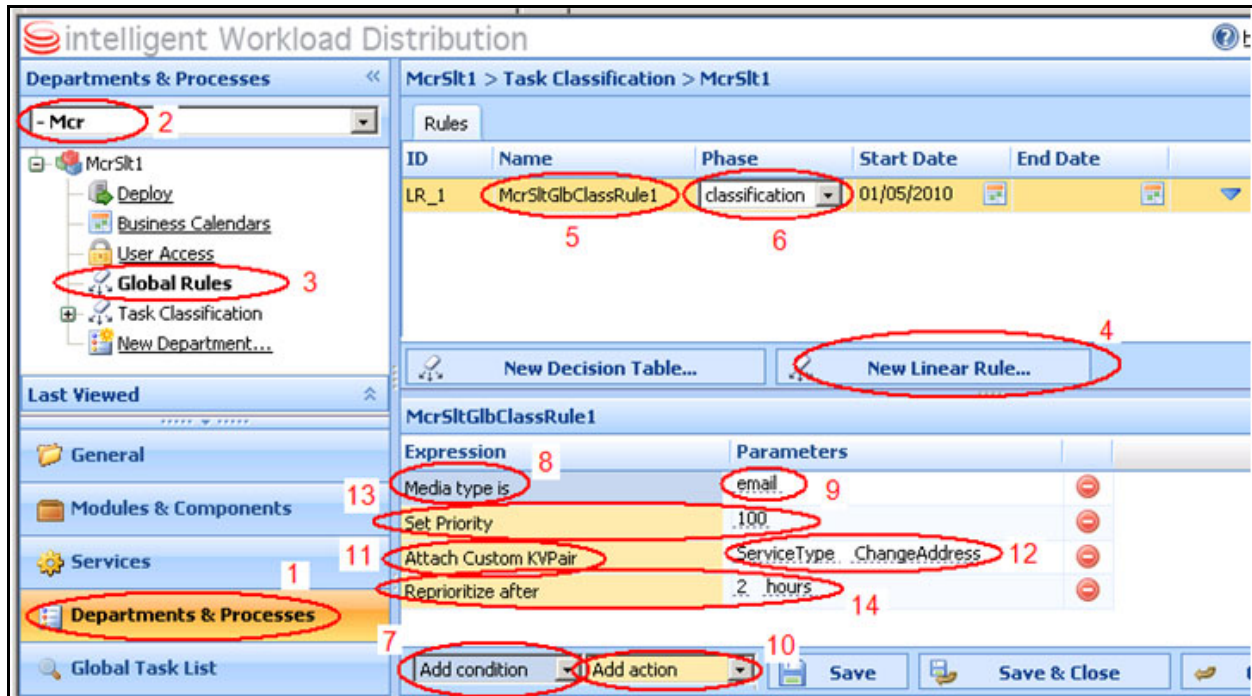


Figure 110: Create a Classification Rule

Create a Prioritization Rule

Create a prioritization rule. The numbers in the following steps are keyed to [Figure 111](#). For simplicity, Global Rules are used in this example, but these prioritization rules could also be created at the Department or Process level. In this case, you would first need to add a rule back at the Global Rules or Capture Point level with the rule action Assign iWD process, to assign the interaction to an iWD Process.

1. Select Departments & Processes.
2. Select your iWD tenant.
3. Select Global Rules.
4. Select New Linear Rule.
5. Enter a name for the new rule.
6. Select prioritization as the Phase.
7. Click Add condition and select a condition.
8. The selected condition is added.

9. Select e-mail.
10. Add one more condition.
11. Select Add action and select Increase Priority and Reprioritize After
12. The selected actions are added.
13. Set values 2 hours and 5 like in Figure 111.
14. Click Save.

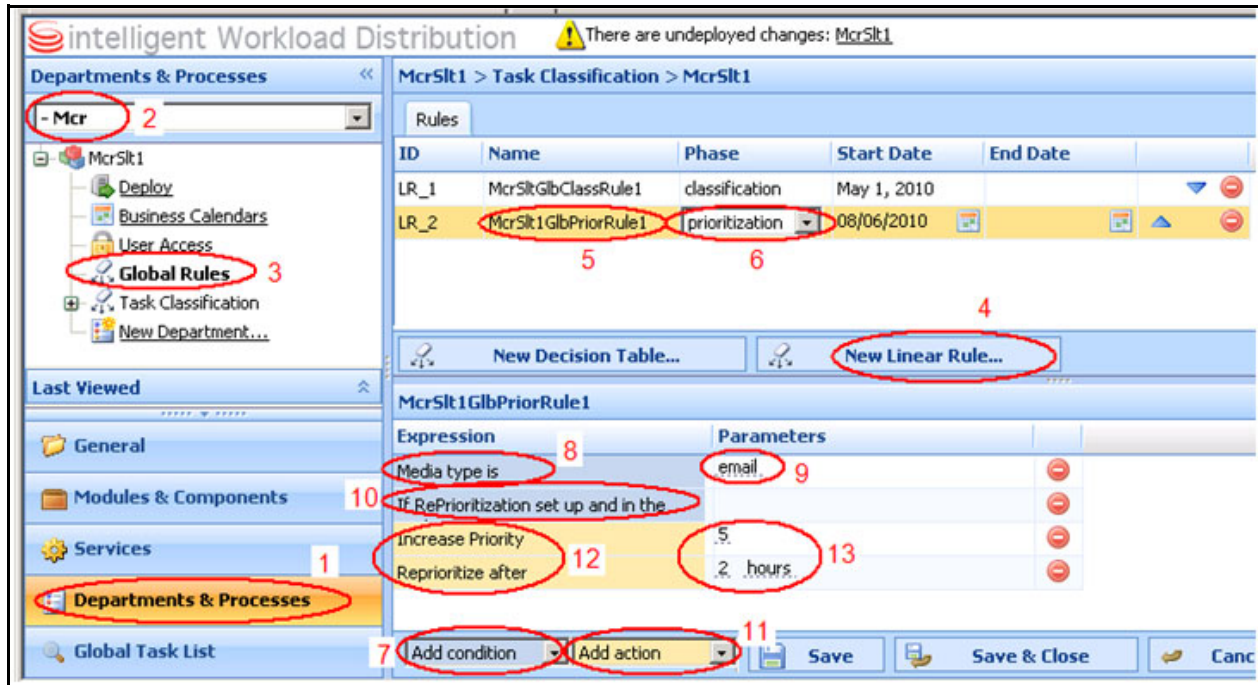


Figure 111: Create a Prioritization Rule

Deploy Modifications

After new rules are created or business objects are modified, the changes have to be deployed before they will have an affect on your business process. If there are undeployed changes in your environment, a notification will appear in the header bar in iWD Manager. See Figure 112. You can either click the solution name (labeled as 1 in Figure 112) or you can click on Deploy under the solution in the navigation tree (labeled as 2 in Figure 112).

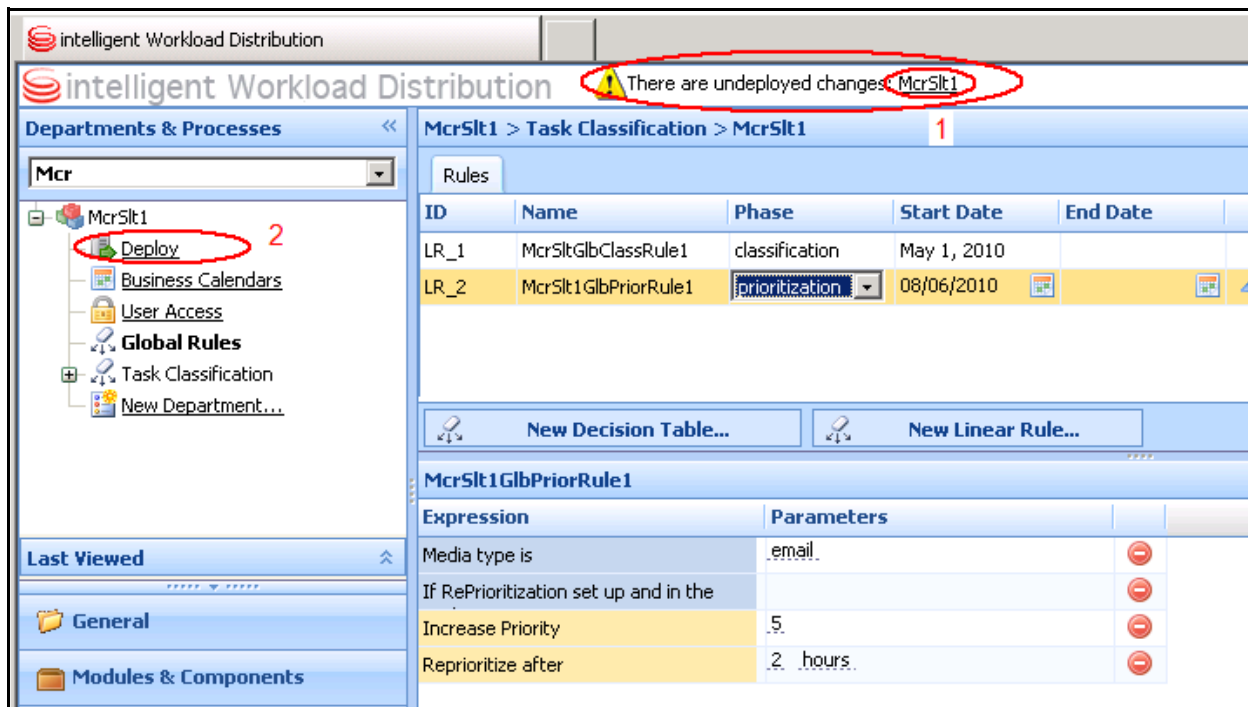


Figure 112: Deploy Changes

Notes on the iWD Business Process

The following are some important things to note about the iWD business process:

- Behavior of the `iWD_Queued` queue and `ActivatedOnly` view—if the interaction does not have a key-value pair with the key `IWD_activationDateTime` it will be processed immediately; otherwise the interaction will be delayed according to the time stamp in the key-value pair.
- Behavior of `iWD_Queued` queue and `ToReprioritize` view—interactions will only be taken through this view if they have a user data key-value pair with the key `IWD_reprioritizeDateTime` and only then if the time that is specified is in the past. This key-value pair is set up by a prioritization business rule. See “Create a Prioritization Rule” on [page 288](#).

Path of E-mail Interactions in IWDBP

All business rules will **only** affect interactions with a `MediaType` equal to `email`. The following are the steps which the interaction will pass through:

- The interaction is submitted by E-mail Server and is placed into the `iWD_New` queue.

- The interaction is processed by the `Classification` strategy. As a result, the classification business rules will be applied to interaction. In this example, `ServiceType` will be set to `ChangeAddress`. Also, the initial `Priority` will be set to `100` and the initial `Reprioritization` time will be set to in 2 hours.
- The interaction is placed into the `iWD_Captured` queue.
- The interaction is processed by the `Prioritization` strategy. As a result, the prioritization business rules will be applied to interaction. In this example we have set the initial priority in classification rules, so prioritization rules will be used on the reprioritization step.
- The interaction is placed into the `iWD_Queued` queue.
- In this example, if no available agents are found, the e-mail interaction will be reprioritized every 2 hours. Its priority will be increased by 5 each time.

Example 2—Departments and Processes

In this example more iWD business objects will be added. This example will show how one flow of interactions can be divided into three streams, and how different business rules can be applied to each stream. In this example the default iWD business process is used for processing Genesys e-mails.

Use Case

In this example the following scenario is used: all interactions with `MediaType = email` should be divided into three groups based on `Subject`: `NewAccount`, `Support`, and all others. All interactions will be delivered to one Agent Group (iWD), but interactions with `Subject = NewAccount` will have highest priority and will be reprioritized more often. Interactions with `Subject = Support` will have lower priority and all other interactions will have the lowest priority and will be reprioritized less often.

Rules Creation

Create a Department and Processes in iWD

Create the Department and Processes in iWD. The following steps are keyed to the numbers in [Figure 113](#).

1. Select `Departments & Processes`.
2. Select your iWD tenant.
3. Select `New Department`.
4. Enter a name for the new department (in this example, `CustomerService`).
5. Click `Save`.

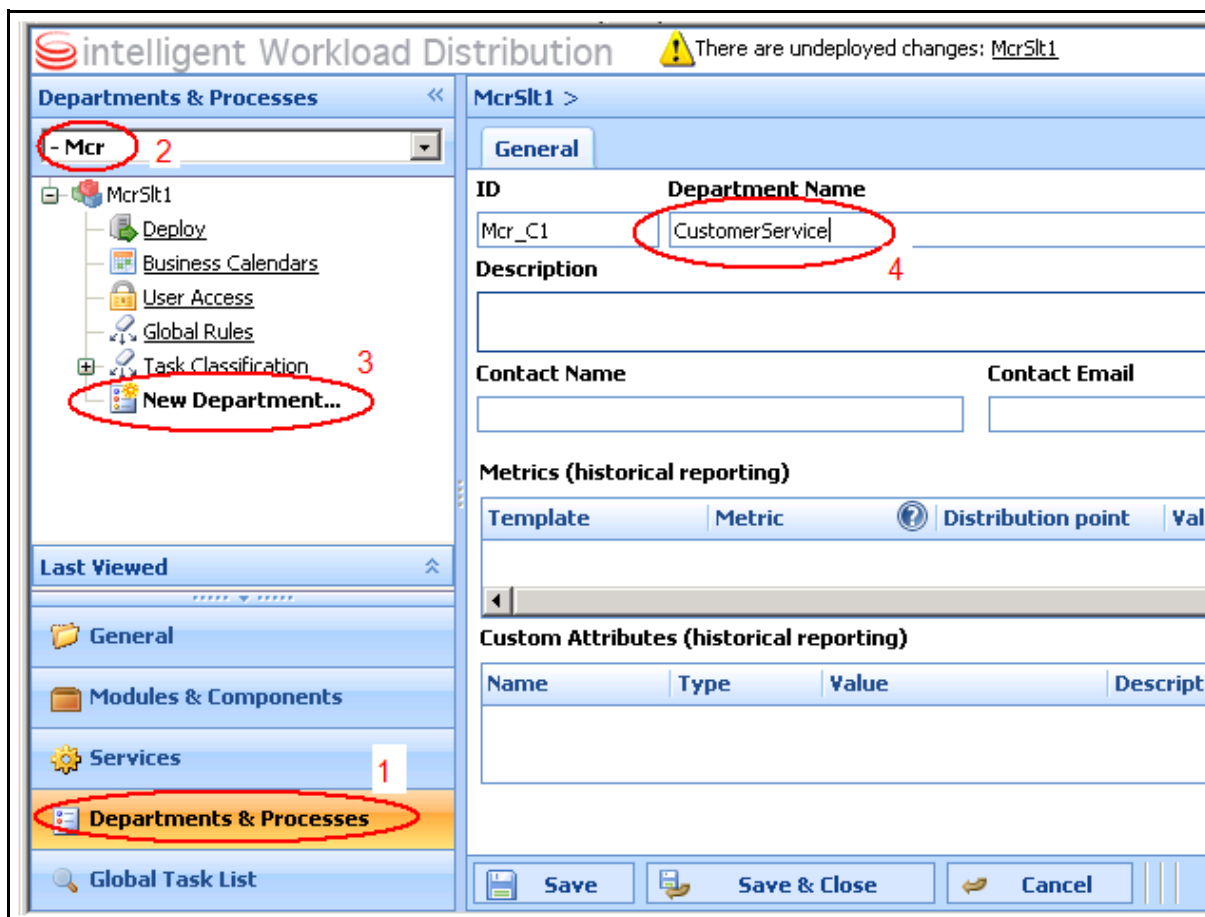


Figure 113: Create Department

Create a new process under the new department. The following steps are keyed to the numbers in [Figure 114](#).

1. Select New Process.
2. Enter a name for the process (in this example, Sales).
3. Click Save.
4. Create two more processes under the CustomerService department: Support and Others.

intelligent Workload Distribution ⚠ There are undeployed changes: McrSlt1

Departments & Processes << McrSlt1 > CustomerService >

General

ID: Mcr_C1_P2 **Process Name:** Sales

Description:

Contact Name: **Contact Email:**

Metrics (historical reporting)

Template	Metric	Distribution point	Value

Custom Attributes (historical reporting)

Name	Type	Value	Description

Figure 114:Create Process

Create Global Classification Rules

Create new Global classification rules as shown in Figure 115, Figure 116, and Figure 117. Remember to save each rule.

intelligent Workload Distribution

Departments & Processes << McrSlt1 > Task Classification > McrSlt1

Rules

ID	Name	Phase	Start Date	End Date
LR_1	McrSlt1GlbClassRule1	classification	01/05/2010	
LR_3	McrSlt1GlbClassRule2	classification	May 1, 2010	
LR_9	McrSlt1GlbClassRule3	classification	May 1, 2010	

New Decision Table... **New Linear Rule...**

McrSlt1GlbClassRule1

Expression	Parameters
Media type is	.email.
If KVPair	.Subject .is .NewAccount.
Assign IWD process	.Sales.

Figure 115:Classification Rules for Example 2—Rule 1

McrSlt1GlbClassRule2		
Expression	Parameters	
Media type is	email	⊖
If KVPair	Subject is Support	⊖
Assign iWD process	Support	⊖

Figure 116: Classification Rules for Example 2—Rule 2

McrSlt1GlbClassRule3		
Expression	Parameters	
Media type is	email	⊖
If KVPair	Subject is not NewAccount	⊖
If KVPair	Subject is not Support	⊖
Assign iWD process	Others	⊖

Figure 117: Classification Rules for Example 2—Rule 3

Create Classification Rules for each Process

Create a new classification rule for each process (use [Figure 118](#) as an example).

For the Sales process:

Action Set Priority 100

For the Support process:

Action Set Priority 50

For the Others process:

Action Set Priority 10

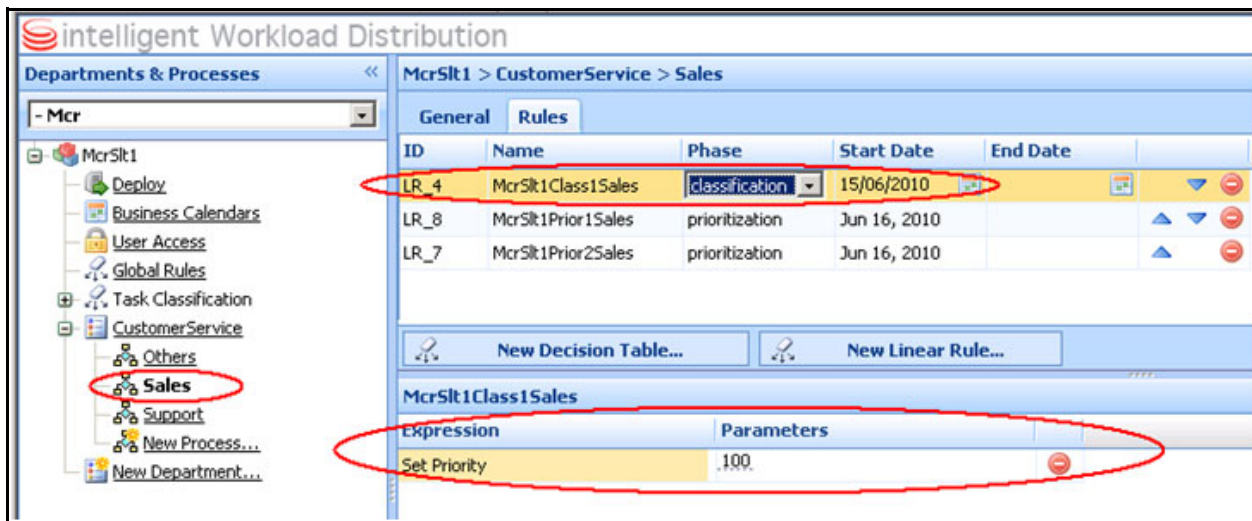


Figure 118: Example Classification Rule for a Process

Create Prioritization Rules for the Processes

Create prioritization rules for your processes. See [Figure 119](#) for an example. For the Sales process, create the rules outlined in [Table 62](#).

Table 62: Sales Process Prioritization Rules

Rule Name	Conditions	Actions
Mcrlt1Prior1Sales	If Reprioritization was not set up	Reprioritize after 1 hour
Mcrlt1Prior2Sales	If Reprioritization was set up and in the past	Reprioritize after 1 hour Increase priority 10

For the Support process, create the rules outlined in [Table 63](#).

Table 63: Support Process Prioritization Rules

Rule Name	Conditions	Actions
Mcrlt1Prior1Support	If Reprioritization was not set up	Reprioritize after 2 hours
Mcrlt1Prior2Support	If Reprioritization was set up and in the past	Reprioritize after 2 hours Increase priority 5

For the Others process, create the rules outlined in [Table 64](#).

Table 64: Others Process Prioritization Rules

Rule Name	Conditions	Actions
McrSlt1Prior1Others	If Reprioritization was not set up	Reprioritize after 3 hours
McrSlt1Prior2Others	If Reprioritization was set up and in the past	Reprioritize after 3 hours Increase priority 1

For rules in this example order is not important, but order can be important for other cases. You can change order of your rules by using the arrow buttons that are circled in [Figure 119](#).

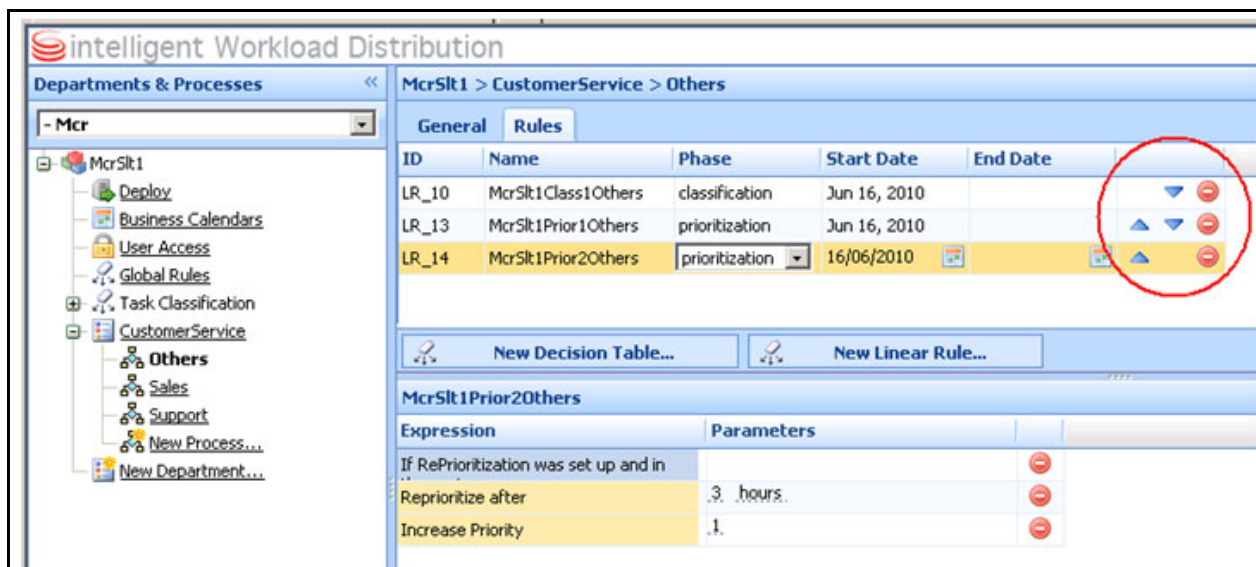


Figure 119: Prioritization Rules for the Others Process

Deploy Changes Deploy your modifications. Refer to [page 289](#).

Path of E-mail Interactions in IWDBP

All business rules will **only** affect interactions with a `MediaType` equal to `email`. The following are the steps which the interaction will pass through:

- The interaction is submitted by E-mail server and is placed into the `iWD_New` queue.
- The interaction is processed by the `Classification` strategy. As a result, the classification business rules will be applied to the interaction. In this example, interaction will be assigned to one of the iWD processes depending on `Subject`.
- Classification rules from the assigned process will be applied. As a result, the initial `Priority` will be set—100 for the Sales process, 50 for the Support process, and 10 for the Others process.

- The interaction is placed into the `iWD_Captured` queue.
- The interaction is processed by the `Prioritization` strategy. As a result, prioritization business rules from the previously assigned process will be applied to the interaction. In this example it means that the interaction will be scheduled for reprioritization (each hour for the `Sales` process, every 2 hours for the `Support` process, and every 3 hours for the `Others` process).
- The interaction is placed into the `iWD_Queued` queue.
- In this example, if no available agents are found, the interaction will be passed into the `Prioritization` strategy based on the schedule that was set up earlier, and the `Priority` will be increased based on the assigned process.

How to Modify IWDBP to Allow Agents to Reply to Inbound E-mails and Send Single Outbound E-mails

This section describes how to modify the iWD business process to allow agents to send replies to inbound e-mails, and to send single outbound e-mails. There are 2 options: create a new queue and a new strategy in IWDBP, or use an existing business process. In this example an existing business process will be used to illustrate how other business processes can be used from the iWD business process.

First, choose a business process to handle agent's replies and outbound e-mails. Second, specify the queue into which the agent's reply will be placed. Third, specify the business process and queue for single outbound e-mails from agents.

In this example the `ABC Simple` business process will process agent's replies (with the `Outbound` queue). Also agents will be given the ability to place interactions into the following queues in IWDBP: `iWD_Completed` and `iWD_Canceled`. All of this will be done in the `Distribution` strategy. The `Route Interaction` properties in the strategy must be as shown in [Figure 120](#).

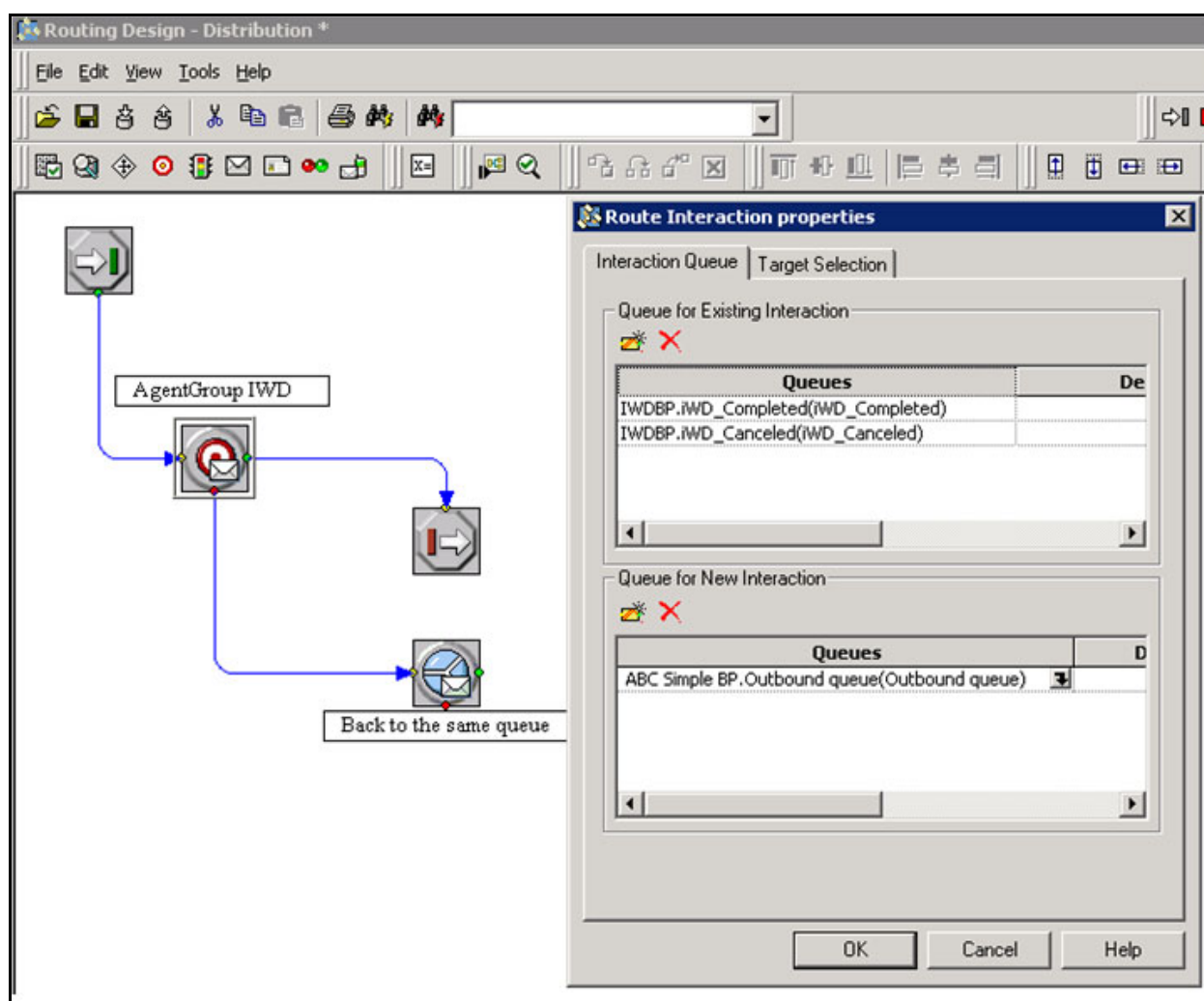


Figure 120:Route Interaction Properties

Next, assign the E-mail Server that will process the outbound e-mail interactions. To do this in the ABC Simple business process, open the Send ABC strategy. In the Send Email property, select the E-mail Server (see [Figure 121](#)).

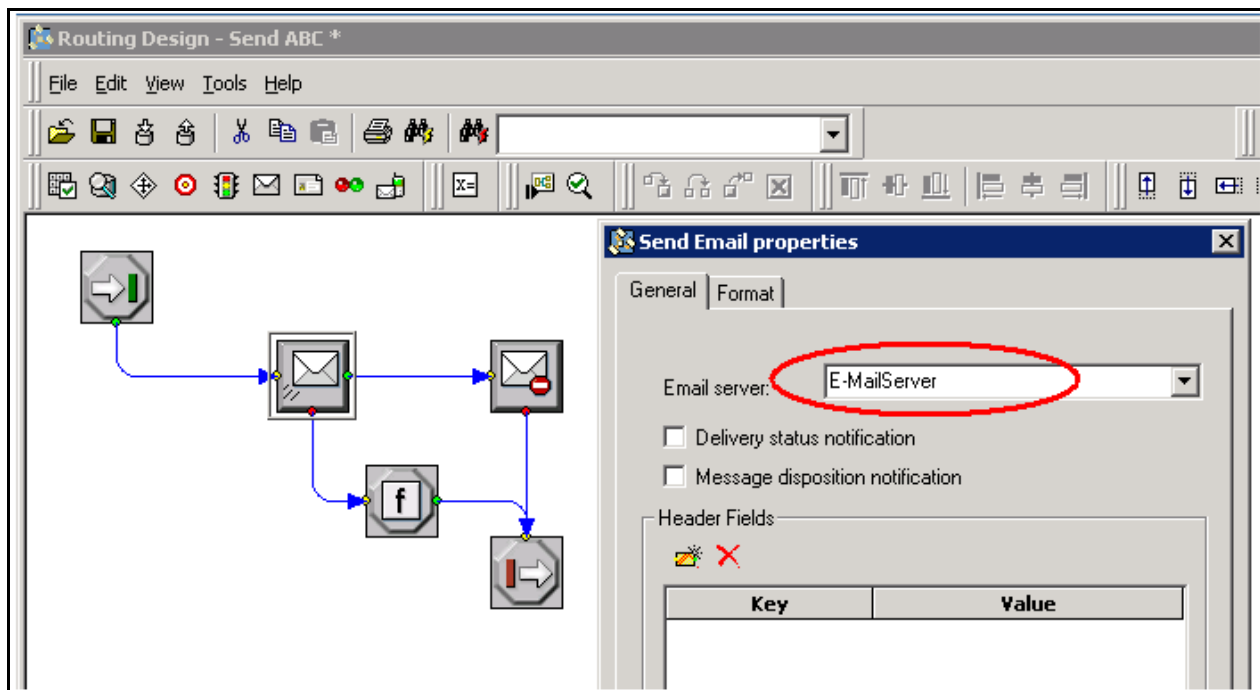


Figure 121: Select E-Mail Server

After these modifications the iWD business process should look like [Figure 122](#).

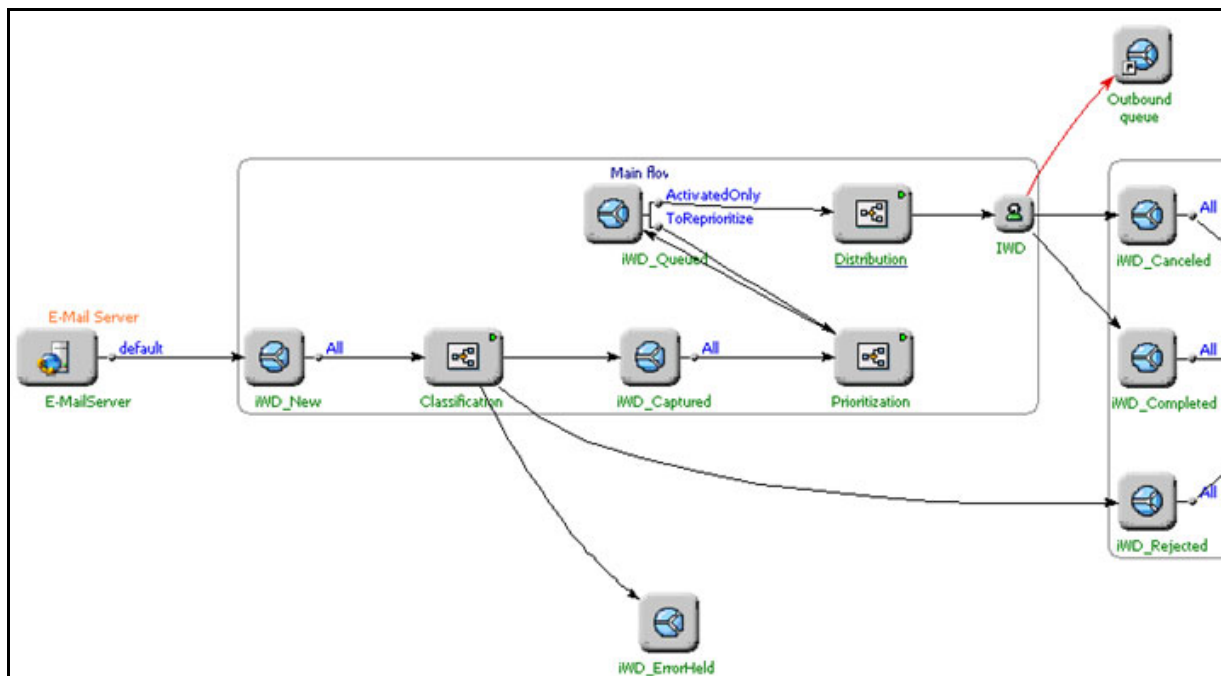


Figure 122: iWD Business Process after Modifications

Save all modifications and run all participating strategies.

Figure 123 shows the ABC Simple business process.

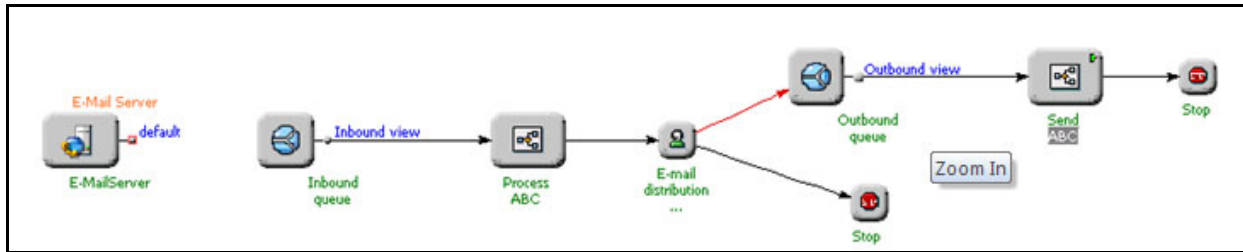


Figure 123:ABC Simple Business Process

Note: In this example only one endpoint is configured for E-mail Server, so there is no connection between E-mail Server and the Inbound queue in the ABC Simple business process.

Example 3—How to Add Another Solution

One iWD business process serves one iWD tenant. One iWD tenant can have several iWD solutions. This example shows how to modify the default iWD business process to make it serve several iWD solutions.

Assume that you want to have two business solutions: one for e-mail interactions (exactly as in Example 2) and a second solution for fax interactions. Fax interactions will be submitted by an XML capture point (Media Type = fax). In this example, the only difference between these two solutions will be the Media Type of interactions.

Environment

There are two streams of input interactions: one from E-mail Server (Media Type = email) and second one from an XML capture point.

Use Case

In this scenario, all interactions from E-mail Server have to pass through business rules that are configured inside of Solution1. All interactions that are submitted by the XML Capture Point have to pass through business rules that are configured inside of Solution2, which is the solution to which this capture point belongs. Generally, any kind of interaction can be passed into the iWD business process, but, due to the rules configuration, business rules will be applied to interactions with Media Type = email (business rules in Solution1) and to interactions with Media Type = fax (business rules in Solution2).

Business rules can be completely different in both solutions, in this example they are created identically, just for simplicity.

Configuration

Create a Solution Create a solution. The following steps are keyed to the numbers in [Figure 124](#).

1. Select your iWD tenant (in this example, Mcr).
2. Select New Solution Instance.
3. Enter a name for the solution (in this example, McrSolution2).
4. Click Save.

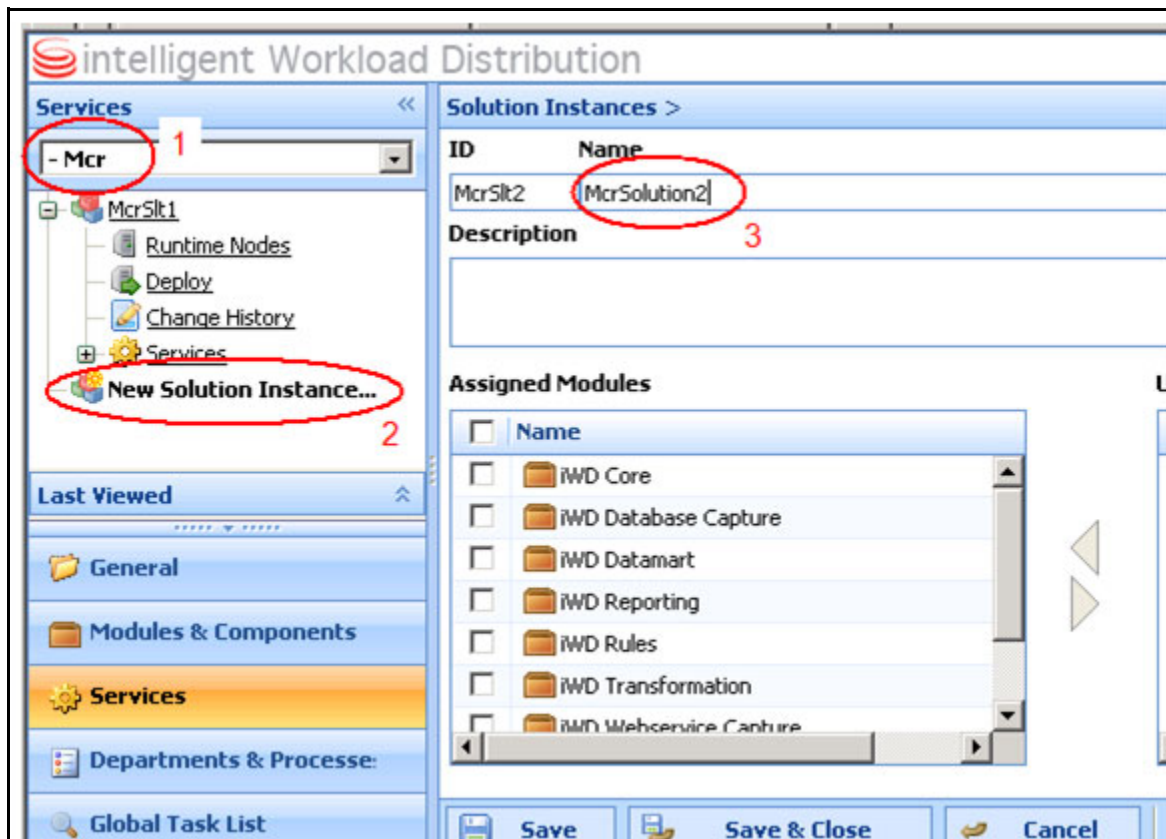


Figure 124: Create a Solution—Example 3

Create Services Create all necessary services under the new solution. See [Figure 125](#).

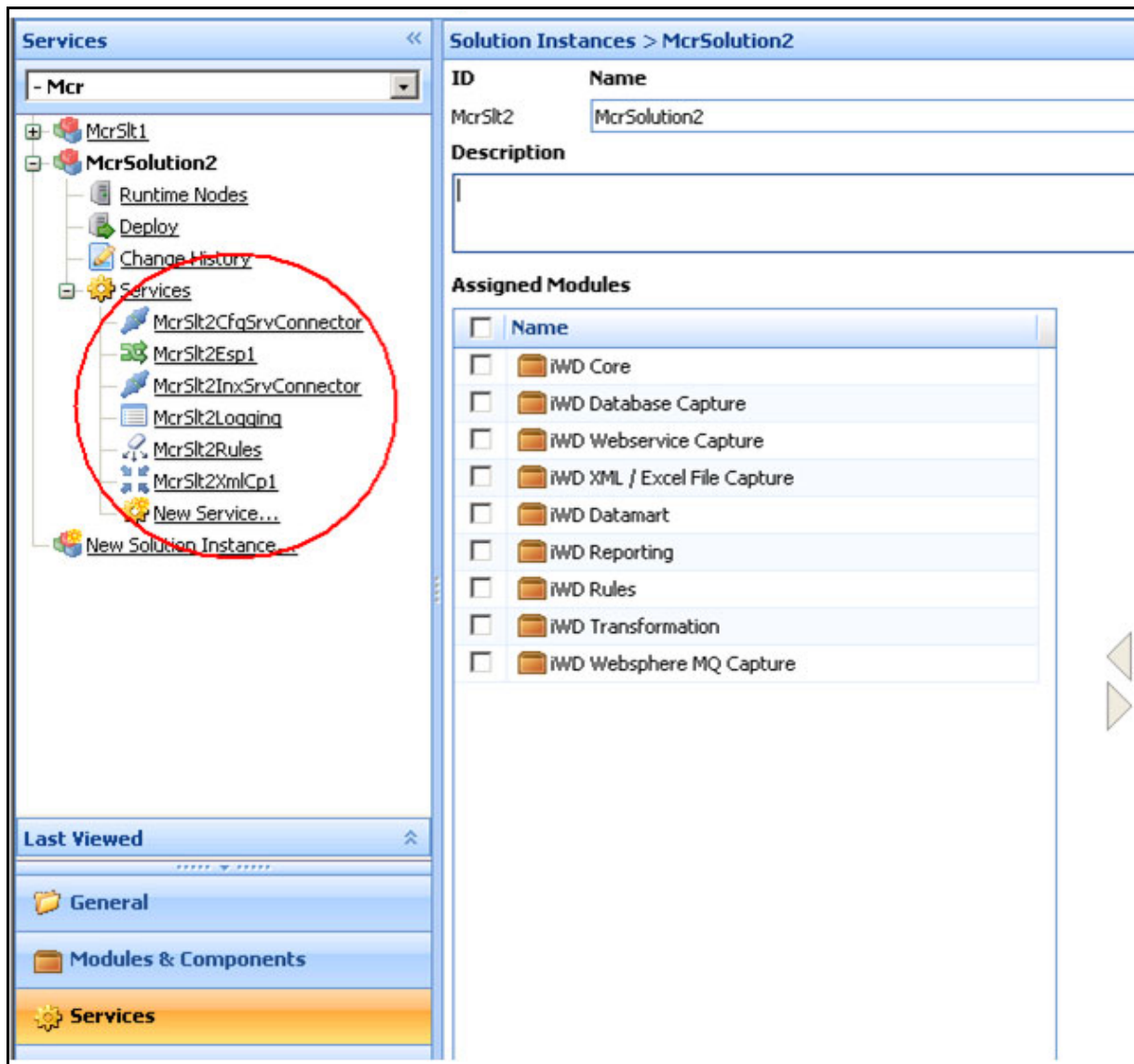


Figure 125: Create Services for the New Solution

Create Department and Processes Create a department and processes as in Example 2 (see “Create a Department and Processes in iWD” on [page 291](#)).

Create Classification Rules Create new Global classification rules as exactly as in Example 2 (see “Create Global Classification Rules” on [page 293](#)), but specify fax as the Media Type.

Create Prioritization Rules Create new prioritization rules as exactly as in Example 2 (see “Create Prioritization Rules for the Processes” on [page 295](#)), but specify fax as the Media Type.

Deploy Modifications Deploy your modifications, as described in “Deploy Modifications” on [page 289](#).

Update the Interaction Server Configuration

Currently iWD supports only one ESP Server per solution. Each ESP Server has to have its own Application configured in Configuration Manager or Genesys Administrator. Therefore, we have to create another Application and add it to the Connections tab of the Interaction Server application. See [Figure 126](#).

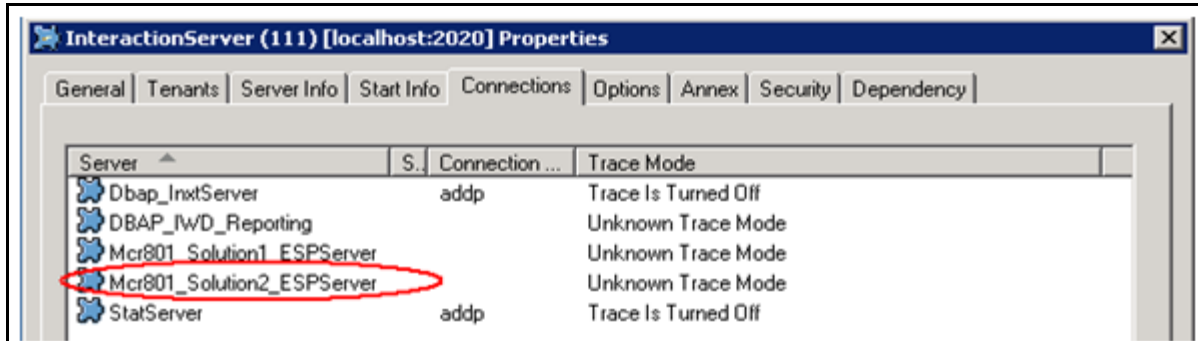


Figure 126: Add ESP Server to Interaction Server's Connections

Modify the Default Business Process

The business process has to be updated to choose the correct ESP Server to invoke from inside the strategy, based on Solution ID. A List Object is used to make a link between the ESP Server and the solution. See “Configuration of List Objects” on [page 252](#) for more information.

After installation the List Object `Iwd_Esp_List` is empty. In this case, Interaction Server will select an ESP Server from one of the Third Party Servers in its Connections tab based on load balancing. Since Example 3 uses two solutions, edit the List Object to reflect the two solutions. The following steps are keyed to the numbers in [Figure 127](#). Open IRD and select the correct tenant, then do the following:

1. Select Routing Design.
2. Select the group List Objects.
3. Select the List Object named `Iwd_Esp_List`.
4. Select item `ESPServiceList`.
5. Enter the Solution ID you attached to the e-mail interaction, as described “Adding Required Properties to Interactions” on [page 275](#). If you did not attach an iWD Solution ID to the e-mail interaction, enter empty as the key.
6. Enter the application name in Configuration Manager (or Genesys Administrator) for the ESP Service from Example2 as a value.
7. Enter the solution ID from Example 3 (`McrSl1t2`) as a key.
8. Enter application name in Configuration Manager (or Genesys Administrator) for the ESP Service from Example 3 as a value (`Mcr801_Solution2_ESPServer`).

9. Save the List Object by clicking OK.

Note: In [Step 5](#), if a Solution ID was not added to the e-mail interaction, empty must be entered as the key because Solution2 is intended to process interactions which were submitted not by an iWD capture point, but by Genesys Media servers (E-mail, Chat, and so on) and therefore do not have a solution ID, unless explicitly added as described in “Adding Required Properties to Interactions” on [page 275](#). The key word empty must be entered exactly as in [Figure 127](#).

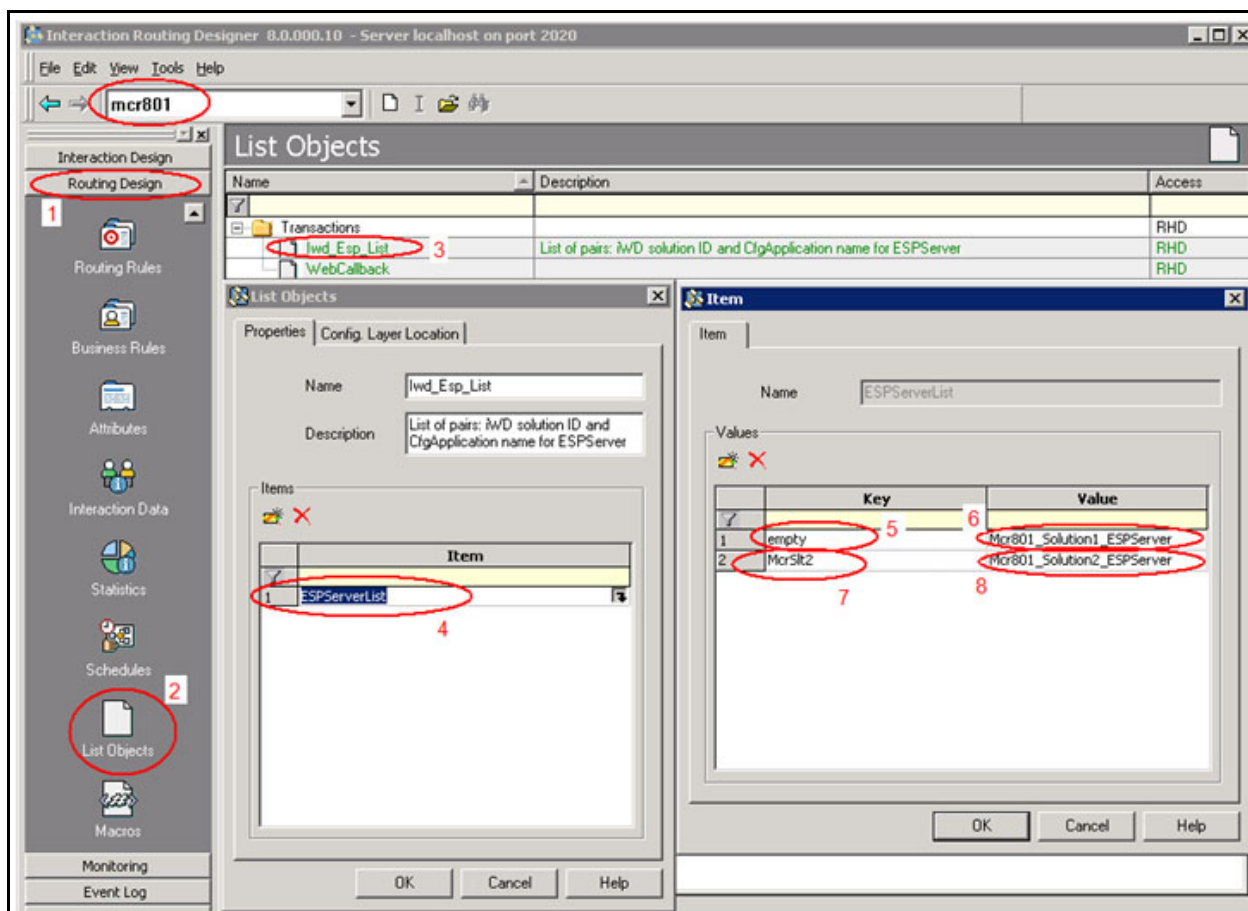


Figure 127: Update the List Object

Path of Interaction

Some important steps are taken in the Classification strategy. As soon as an interaction is placed in the strategy a decision is made as to which ESP Server to invoke for this interaction. This decision is made based on the presence and value of the key-value pair IWD_solutionId in the user data of the

interaction. Interactions that were created by iWD capture points will have this key-value pair and the Solution ID will equal the ID of the Solution to which this Capture Point belongs.

- If the key-value pair is missing from user data, then the ESP Server will be taken from the List Object `Iwd_Esp_List`, item `ESPServerList`, value for key empty (all interactions submitted by E-mail server will not have the key-value pair, and therefore in this example `Mcr801_Solution1_ESPServer` will be chosen).
- If the key-value pair is found and matches with a key in List Object `Iwd_Esp_List`, item `ESPServerList`, then the corresponding ESP Server will be chosen (in this sample each interaction submitted by XML Capture Point will have key-value pair `IWD_solutionId = McrSlr2`, and therefore ESP Server `Mcr801_Solution2_ESPServer` will be chosen for these interactions).
- If the key-value pair `IWD_solutionId` is found but does not match with any key in List Object `Iwd_Esp_List`, item `ESPServerList`, then the ESP Server will be chosen by Interaction Server and the result is unpredictable (Interaction Server will select one of Third Party Servers in its `Connections` tab based on load balancing).

Path of Interactions Submitted by E-mail Server

The following flow is used for interactions that are submitted by E-mail Server (`MediaType = email`):

- The interaction is submitted by E-mail Server and is placed into the `iWD_New` queue.
- The interaction is processed by the `Classification` strategy. ESP Server `Mcr801_Solution1_ESPServer` will be assigned and invoked. As a result, classification business rules will be applied to the interaction. In this example, the interaction will be assigned to one of the iWD business processes depending on `Subject`.
- Classification rules from the assigned process will be applied. As a result, the initial `Priority` will be set—100 for the `Sales` process, 50 for the `Support` process and 10 for the `Others` process.
- The interaction is placed into the `iWD_Captured` queue.
- The interaction is processed by the `Prioritization` strategy. As a result, prioritization business rules from the previously assigned process will be applied to the interaction. In this example, that means that the interaction will be scheduled for reprioritization (each hour for the `Sales` process, every 2 hours for the `Support` process, and every 3 hours for the `Others` process).
- The interaction is placed into the `iWD_Queued` queue.

- In this example, if no available agents are found, the interaction will be passed to the `Prioritization` strategy based on the schedule that was set up earlier and the `Priority` will be increased based on assigned process.

Path of Interactions Submitted by a Capture Point

As was stated earlier, the two solutions are identical, and the only difference is that `Solution1` will apply business rules to e-mail interactions, and `Solution2` will apply business rules to fax interactions. Therefore the following flow is the same as above for e-mail interactions except for the ESP Server. You can modify business rules for these interactions however you wish.

The following flow is used for interactions that are submitted by a Capture Point (`MediaType = fax`):

- The interaction is submitted by an XML Capture Point and is placed into the `iWD_New` queue.
- The interaction is processed by the `Classification` strategy. `ESP Server Mcr801_Solution2_ESPServer` will be assigned and invoked. As a result, classification business rules will be applied to the interaction. In this example, the interaction will be assigned to one of the iWD business processes depending on `Subject`.
- Classification rules from the assigned process will be applied. As a result, the initial `Priority` will be set—100 for the `Sales` process, 50 for the `Support` process and 10 for the `Others` process.
- The interaction is placed into the `iWD_Captured` queue.
- The interaction is processed by the `Prioritization` strategy. As a result, prioritization business rules from the previously assigned process will be applied to the interaction. In this example, that means that the interaction will be scheduled for reprioritization (each hour for the `Sales` process, every 2 hours for the `Support` process, and every 3 hours for the `Others` process).
- The interaction is placed into the `iWD_Queued` queue.
- In this example, if no available agents are found, the interaction will be passed to the `Prioritization` strategy based on the schedule that was set up earlier and the `Priority` will be increased based on assigned process.



Appendix

D

iWD Rules and Existing Business Processes

This Appendix explains how to use iWD business rules functionality with existing business processes. It explains what modifications have to be done in order to use iWD business rules within existing business processes.

This Appendix contains the following sections:

- [Using iWD Rules in Existing Business Processes, page 307](#)
- [Example, page 309](#)

Using iWD Rules in Existing Business Processes

Remember that because you are not using the standard IWDBP, if you view these interactions in the Global Task List, sometimes the Status column will be empty. This is because the Status reported through the Global Task List is taken from the name of the queue in which the interaction is placed within the Business Process, and the Global Task List assumes that these queues all have names beginning with iWD_. Therefore, when the interaction is located in any other queue, such as Inbound in this example, it will not be displayed in the Status column.

Requirements:

- A Genesys E-mail solution is installed. (An e-mail solution is used in the example. Modifications will be the same for any other type of media).
- An iWD solution is installed.

Assumptions:

- There is only one iWD solution per business process. If you want to use a business process in several iWD solutions, you must:
 - For iWD native interactions (which always have `IWD_solutionId`):
 - change the List Object `Iwd_Esp_List` accordingly (see “Configuration of List Objects” on [page 252](#)).

Note: “Native interactions” refers to interactions captured by an iWD capture adapter. Interactions going through the iWD Business Process that do not come through an iWD capture adapter (that is, interactions coming into the system from a standard Genesys media server, through a Gplus Adapter, or through an integration built with the Genesys Open Media SDK) are referred to as “non-native” or “foreign” interactions.

- For non-native interactions (which do not have `IWD_solutionId`):
 - change the logic for assigning `solutionId`, based on an interaction’s property, in the `IWD_BusinessRules_Ext` strategy (Assign Properties block with comment `solutionId = 'Your_solution_id'`).
Figure 133 on [page 311](#) shows the strategy.
- Interactions with `MediaType = email` are the only interactions that come as input into existing business processes. If you expect interactions of several media types as input, you must adjust classification rules in iWD Manager accordingly (add condition “Media type is”).
- There is only one Agent Group to which the interactions can be assigned. If you want to use several Agent Groups you must modify the target selection in the Process ABC strategy, which is part of the ABC Simple BP business process that comes with eServices Interaction Workflow installation.

ABC Simple BP

The example uses the ABC Simple BP business process that comes with eServices Interaction Workflow installation. See [Figure 128](#).

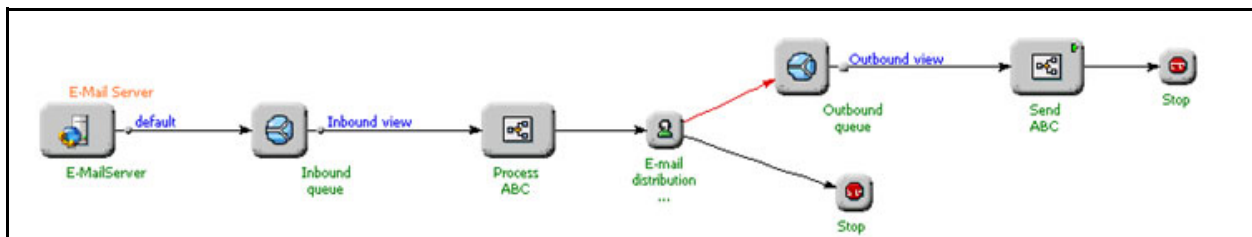


Figure 128:ABC Simple BP

Example

This is a simple example that shows how a business process could be modified to use iWD rules.

Use Case

In this scenario, there is a working business process, and you want to deliver interactions to agents based on priority. Priority should be assigned based on e-mail subject.

All incoming interactions (in this example they all have `MediaType = email`) should be divided into three groups based on `Subject`: `NewAccount`, `Support`, and all others. All interactions will be delivered to one Agent Group (IWD), but interactions with `Subject = NewAccount` will have highest priority and will be reprioritized more often. Interactions with `Subject = Support` will have lower priority and all other interactions will have the lowest priority and will be reprioritized less often.

Modify the Existing Business Process

Modify the existing business process. Add one more strategy, which will invoke the iWD service to apply business rules. Also, one queue will be added to the business process. This queue will provide the mechanism for reprioritization and delivering interactions to an agent based on priority.

The iWD Setup Utility includes a sample business process that contains a couple of strategies and a queue. [Figure 129](#) shows the ABC IWD Simple BP business process.

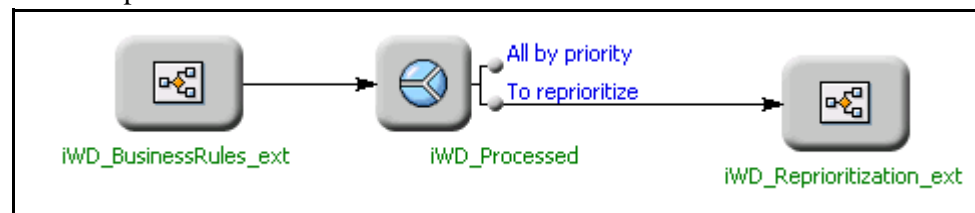


Figure 129:ABC IWD Simple BP

In this business process, the property of the All by priority view is configured as shown in [Figure 129](#) (the Conditions tab is empty). Thus,

interactions from this queue and through this view will be taken by priority. The interaction with the highest priority will be taken first.

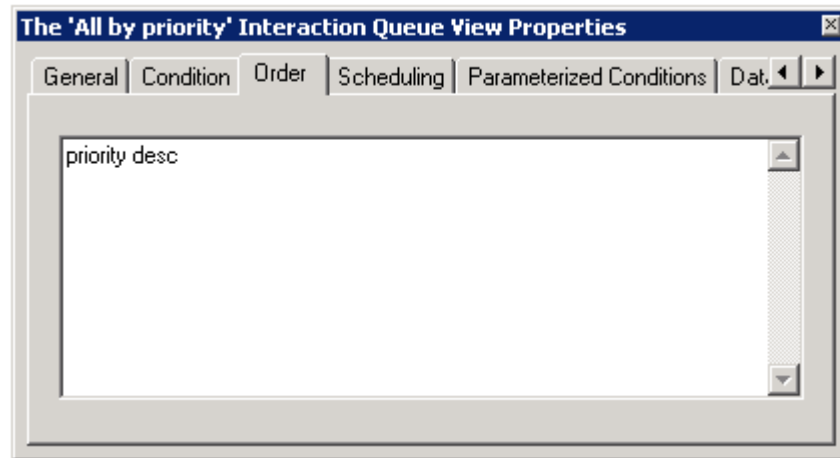


Figure 130:All by Priority Interaction Queue View Properties

Properties of the To reprioritize view are configured as in [Figure 131](#) and [Figure 132](#). Thus, interactions from this queue and through this view will be taken sorted by IWD_reprioritizeDateTime; and only if IWD_reprioritizeDateTime was set and has expired.

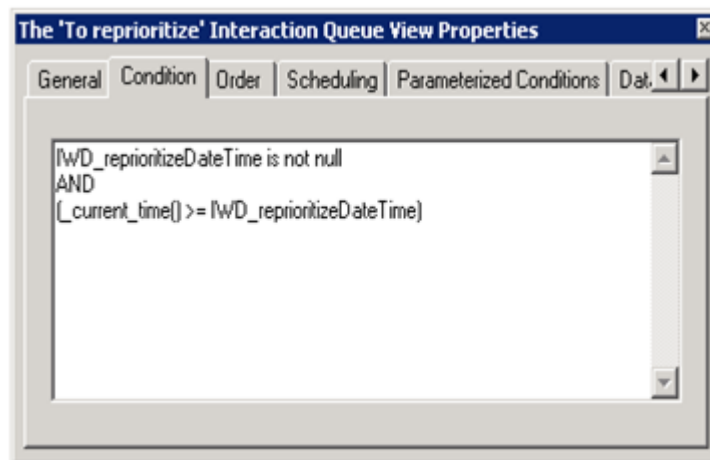


Figure 131:'To Reprioritize' Properties—Condition Tab

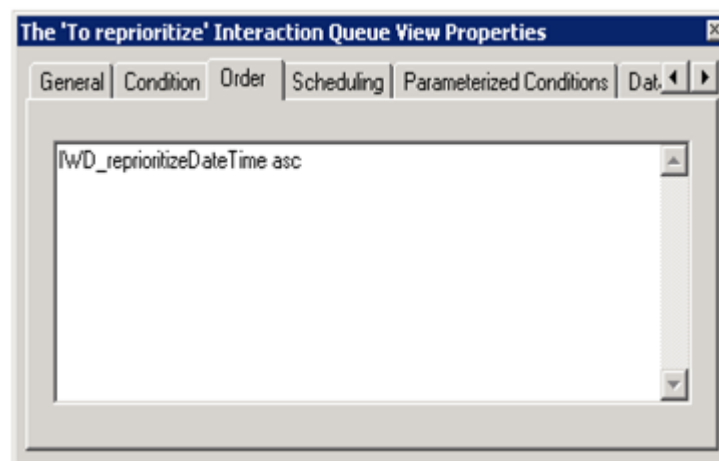


Figure 132: 'To Reprioritize' Properties—Order Tab

Figure 133 shows the iWD Business Rules Ext strategy.

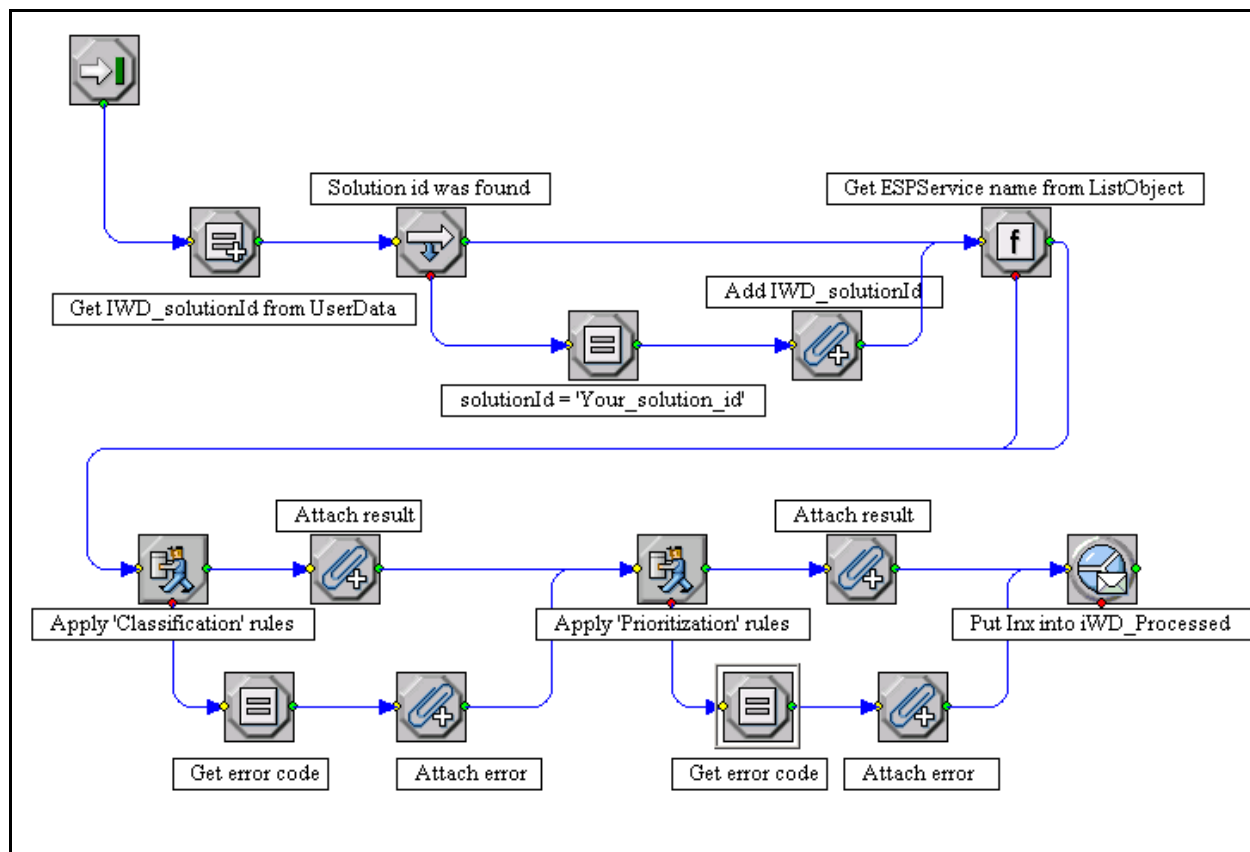


Figure 133: iWD Business Rules Ext Strategy

Figure 134 shows the iWD_Reprioritization_Ext strategy. The key-value pair IWD_solutionId will always be attached **after** the IWD_BusinessRules_Ext

strategy. This is why no check is made for the presence of that key-value pair in the strategy.

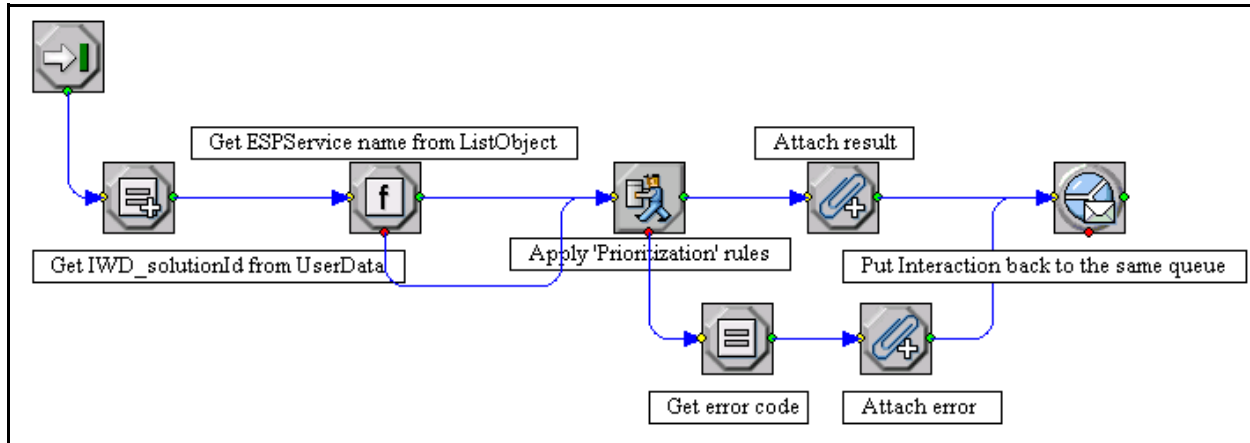


Figure 134:iWD Reprioritization Ext Strategy

Move the iWD_Processed queue, as well as the iWD_BusinessRules_Ext and iWD_Reprioritization_Ext strategies from the ABC IWD Simple BP business process to the ABC Simple BP business process. Figure 135 shows how the ABC Simple BP will look at this point.

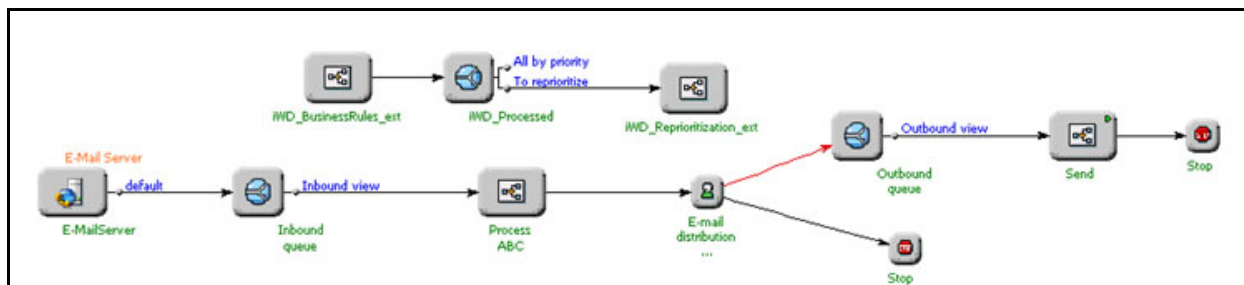


Figure 135:ABC Simple BP with Queue and Strategies Added

Insert the added group between Inbound queue and Process ABC. Figure 136 shows how the ABC Simple BP business process will look at this point.

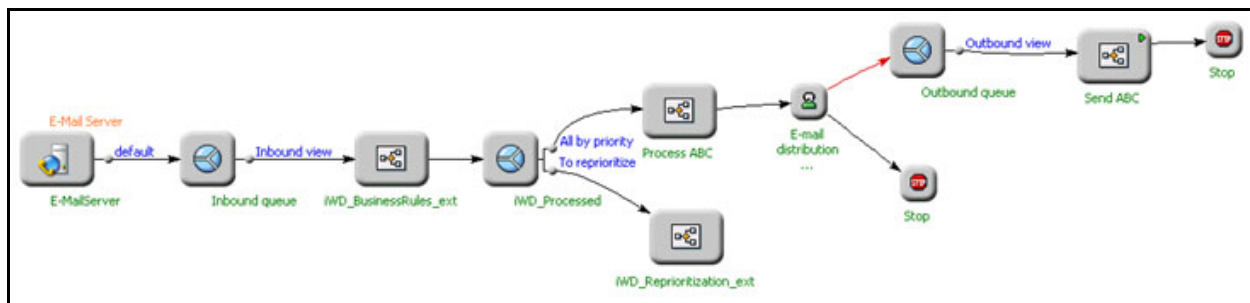


Figure 136:ABC Simple BP with Group Added

The existing business process is now updated. Next, create business rules in iWD.

Rules creation

Open iWD Manager and modify the Standard Rules Template as described in “Modify the Standard Rules Template” on [page 285](#). Assume that all incoming interactions have `MediaType = email`, so you only need to add the four new Actions.

Create an iWD Tenant and an iWD Solution, as described in “iWD Configuration” on [page 284](#).

Create a department and processes as described in “Create a Department and Processes in iWD” on [page 291](#).

Create new Global classification rules as described in “Create Global Classification Rules” on [page 293](#) (in this example we do not check media type, assuming that all interactions have `MediaType = email`).

Create prioritization rules for your processes as described in “Create Prioritization Rules for the Processes” on [page 295](#).

Deploy your modifications.

Path of E-mail Interactions

The following are the steps which the interaction will pass through:

- The interaction is submitted by E-mail server and is placed into the Inbound queue.
- The interaction is processed by the `IWD_BusinessRules_Ext` strategy:
 - The `SetBusinessContext` method of the ESP Service is invoked:
 - First, global classification rules are applied. As a result, the interaction will be assigned to one of iWD processes depending on `Subject`.
 - Immediately after global classification rules are applied, classification rules of the assigned Department and Process will be applied. As a result, the initial `Priority` will be set—100 for the Sales process, 50 for the Support process, and 10 for the Others process.
 - The Classification step is completed.
 - The `Prioritize` method of the ESP Service is invoked:
 - As a result, prioritization business rules from the previously assigned process will be applied to the interaction. In this example, that means that the interaction will be scheduled for reprioritization (each hour for the Sales process, every 2 hours for the Support process, and every 3 hours for the Others process).
 - The Prioritization step is completed.
- The `IWD_BusinessRules_Ext` strategy is completed and the interaction is placed into the `iWD_Processed` queue.

- In this example, if no available agents are found, the interaction will be passed to the `IWD_Reprioritization_Ext` strategy based on the schedule that was set up earlier. Priority will be increased based on the prioritization rules specified in the assigned process. Interaction will be rescheduled for reprioritization and placed back into the `iWD_Processed` queue.



Supplements

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)

- *iWD Data Mart Reference Manual*, which describes/provides information about the iWD Data Mart.
- *iWD WebSphere MQ Capture Adapter Reference Guide*, which provides installation and configuration information for the WebSphere MQ Capture Adapter.

In addition, the <https://sites.google.com/a/iwdlab.com/iwd8/> website provides additional information about working with iWD. The content on this website is updated frequently.

eServices

- *eServices 8.0 Deployment Guide*, which includes a high-level overview of features and functions of Genesys eServices together with architecture information and deployment-planning materials. It also introduces you to some of the basic concepts and terminology that are used in this product. The *eServices Deployment Guide* also provides information about deploying the JMS Integrated Capture Point.
- *eServices 8.0 User's Guide*, which provides overall information and recommendations on the use and operation of Genesys eServices, and the use and operation of the JMS Integrated Capture Point.
- *eServices 8.0 Reference Manual*, which provides information about the configuration options for each eServices component, including options specific to the JMS Integrated Capture Point.
- *eServices 8.0 Open Media Interaction Models Reference Manual*, which presents a set of basic interaction models—showing the components that are involved and the messaging (requests, events) among them.

- “eServices Log Events” in the *Framework 8.0 Combined Log Events Help*, which is a comprehensive list and description of all events that may be recorded in Management Layer logs.

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 for the Genesys 8.x releases.
- *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 that are relevant to the Genesys licensing system.
- *Genesys Database Sizing Estimator 8.x Worksheets*, which provides a range of expected database sizes for various Genesys products.

For additional system-wide planning tools and information, see the release-specific listings of System Level Documents on the Genesys Technical Support website. These documents are accessible from the [system level documents by release](#) tab in the Knowledge Base Browse Documents Section.

Genesys product documentation is available on the:

- Genesys Technical Support website at <http://genesyslab.com/support>.
- Genesys Documentation wiki at <http://docs.genesyslab.com/>.

Related Documentation Resources

- Genesys Documentation Library DVD and/or the Developer Documentation CD, which you can order by e-mail from Genesys Order Management at orderman@genesyslab.com.

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 65](#) describes and illustrates the type conventions that are used in this document.

Table 65: Type Styles

Type Style	Used For	Examples
Italic	<ul style="list-style-type: none">Document titlesEmphasisDefinitions of (or first references to) unfamiliar termsMathematical 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 319).	Please consult the <i>Genesys Migration Guide</i> for more information. Do <i>not</i> use this value for this option. A <i>customary and usual</i> 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 65: Type Styles (Continued)

Type Style	Used For	Examples
Monospace font (Looks like teletype or typewriter text)	<p>All programming identifiers and GUI elements. This convention includes:</p> <ul style="list-style-type: none"> The <i>names</i> 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. <p>Also used for any text that users must manually enter during a configuration or installation procedure, or on a command line.</p>	<p>Select the Show variables on screen check box.</p> <p>In the Operand text box, enter your formula.</p> <p>Click OK to exit the Properties dialog box.</p> <p>T-Server distributes the error messages in EventError events.</p> <p>If you select true for the inbound-bsns-calls option, all established inbound calls on a local agent are considered business calls.</p> <p>Enter exit on the command line.</p>
Square brackets ([])	<p>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.</p>	<pre>smcp_server -host [/flags]</pre>
Angle brackets (< >)	<p>A placeholder for a value that the user must specify. This might be a DN or a port number specific to your enterprise.</p> <p>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.</p>	<pre>smcp_server -host <confighost></pre>



Index

Symbols

[] (square brackets)	319
< > (angle brackets)	319

A

ABC IWD Simple BP business process	309
ABC Simple BP business process	308
Access Groups	116
ACME Sample Application	94
Actions	
properties and actions	130
ADDP	56–58
angle brackets	319
Apache	41
application cluster	222
application installation	28
Archive strategy	254, 269
audience, for document	13

B

Boolean	226
BPR_iWD_Extension.jar	173
BPR_iWD_Extension.jar Stat Server option	78
brackets	
angle	319
square	319
Business Attribute	71
Business Calendars view	183
Business Context Management	
Service	74, 80, 151
creating	80
high availability	221
business logic	182

C

CancelTask message	243
capacity rule	68, 71
Capture Adapters	17
Database	18
JMS	18, 42, 60, 147
Web Service	17
XML	17
CCPulse+ template	179
centralized logging	21
centralized logging configuration	82
Change History view	136, 144
properties and actions	145
Classification strategy	253, 257
CME Tenant	113
commenting on this document	13
compatibility	41
completed-queues Interaction Server option	85
CompleteTask message	239
component	125
Conditions	
properties and actions	130
configuration sections	106
Departments & Processes	106
Global Task List	106
Modules & Components	106
Services	106
Configuration Server	101
Configuration Server Connector	
Service	80, 150, 153
creating	80
configuring	
a custom interaction property	249
centralized logging	82
iWD Manager logging	82
iWD Runtime Node logging	82
using the iWD Setup Utility	69
Context history	109
conventions	
in document	318

type styles 318
 CreateTask message 228
 creating
 iWD Configuration database 50
 iWD Data Mart database 62
 iWD Runtime Node in iWD Manager 65
 iWD Solution in iWD Manager 65
 iWD Tenant in iWD Manager 65
 services in iWD Manager 80
 current-day statistics 179
 custom properties (interaction) 248

D

Data Mart ETL services 173
 Data Types 226
 database
 iWD Configuration database 28
 Database Capture Adapter 18
 Database Capture Point 161
 properties 162
 Database Capture Point Service 151
 database preparation 28
 Database Service 151, 169
 properties 169
 databases
 Interaction Server 85
 DateTime 226
 Department Details view 182, 183
 Departments & Processes
 configuration section 106, 182
 Deployment
 Departments and Processes 183
 properties and actions 143
 Solutions and Services 136, 143
 Details view 103, 107
 DetermineESPServerName subroutine 254
 Direction 225
 Distribution strategy 254, 263, 279
 document
 audience 13
 change history 14
 conventions 318
 errors, commenting on 13
 version number 318

E

Error message 230
 ESP block 25
 ESP object 222
 Extended Statistics Service 170
 External Service Protocol (ESP) block 25

F

font styles
 italic 318
 monospace 319
 Functions tab 133
 properties and actions 134

G

General configuration 111
 Genesys Administrator 101
 Genesys Chat 12
 Genesys E-mail 12
 Genesys Info Mart 21
 Genesys Interaction Server Connector
 Service
 properties 155
 Genesys SMS 12
 GetTaskInfo message 231
 Global Rules 192
 Global Task List 106
 configuration section 106

H

Header bar 103, 104
 historical reporting 21
 History view 123
 properties and actions 124
 HoldTask message 240

I

ICON 21
 ics_multi_media_agent_seat technical
 license 22
 Import and Export function 114
 Import and Export view 111
 Import/Export properties and actions 115
 Import/Export view 114
 In messages 225
 installation 27
 installing
 Interaction Server and its databases 43
 iWD Capture 60
 iWD Data Mart 61
 iWD Manager 51
 iWD Rules 59
 iWD Runtime Node 46
 iWD Setup Utility 68
 iWD Stat Extensions 78
 Integer 226

intelligent Workload Distribution
 features 17
 overview 17
 intended audience 13
 Interaction Concentrator 21
 interaction custom properties 248
 interaction properties mapping 245
 Interaction Queues 254
 Interaction Server 41
 Interaction Server Connector
 Service. 80, 150, 154
 creating 80
 Interaction Server database 86
 interactions table 245, 248
 italics 318
 iWD
 system configuration 29
 Configuration database 28
 features 17
 installation 27
 overview 17
 iWD application. 12
 iWD business process 251
 iWD business rules
 using within an existing business process. 307
 iWD CDs 41
 iWD Extended Statistics Service 170
 iWD installation
 preparation 39
 prerequisites. 40
 iWD Manager. 97
 iWD Rules Service 25
 iWD services
 recommended configuration order 150
 iWD Setup Utility
 installing 68
 iWD solution 12
 iWD state model 25
 IWD_BusinessRules_Ext strategy 311
 iWD_Canceled queue 254
 iWD_Captured queue 254
 iWD_Completed queue 254
 iWD_ErrorHeld queue 254
 Iwd_Esp_List 252
 iWD_New queue 254
 iWD_Queued queue 254
 iWD_Rejected queue. 254
 iWD_Reprioritization_Ext strategy 311
 IWDBP 251
 adapting for standard Genesys channels . 275
 strategies 253

J

Java SDK 41
 java-extension-jar Stat Server option 78
 jaxb-api.jar 48, 55
 jaxb-impl.jar 48, 55
 jdbc-driver Stat Server option 79
 jdbc-url Stat Server option 79
 JMS Capture Point. 18, 42, 60, 147, 157

K

Kettle ETL Service. 151, 173
 properties 174

L

Last Viewed 107
 licensing 21
 List Object 252
 Log Viewer 145
 log4j.properties 83
 Logging Service 80, 150, 152
 creating 80
 properties 152
 logs
 Management Layer 316
 lookup tables. 120
 Lookup Tables view 111

M

managed tenant 29
 Managed Tenant Details view 111, 118
 Managed Tenant view
 properties and actions 119
 Management Layer logging 316
 Message Server 21
 messages
 CancelTask 243
 CompleteTask. 239
 CreateTask 228
 Error. 230
 GetTaskInfo 231
 HoldTask 240
 Ping 244
 Pong 244
 RestartTask 242
 ResumeTask 241
 Task Action 226
 Task Notification 227
 TaskAssigned 239
 TaskCanceled 243
 TaskCompleted 239
 TaskCreated 230

- TaskDistributedQueue 238
- TaskErrorHeld 241
- TaskHeld 240
- TaskInfo 231
- TaskRejected 244
- TaskRestarted 242
- TaskResumed 241
- TaskUpdated 237
- UpdateTask 236
- Metric template
 - value types 135
- Metrics templates 125, 135
- Modules 126
- Modules & Components
 - configuration section 106
- Modules view
 - properties and actions 127
- monospace font 319

N

- native interactions 308
- Navigation panel 104
- navigation sections
 - General 106
- navigation tree 107
- navigation sections 106
- Navigational panel 103

O

- Object Actions 108
- Open Media API 12
- Out messages 225

P

- Parameters tab 131
 - properties and actions 131
- password encoder 53
- password Stat Server option 79
- passwordEncoder.cmd file 53
- Person 101
- Ping message 244
- Pong message 244
- Prioritization strategy 253, 260
- Process Details view 182
- Profile view 111, 112
 - properties and actions 112

R

- refresh-interval Stat Server option 79
- Reporting 21

- RestartTask message 242
- ResumeTask message 241
- Rule Template view
 - properties and actions 129
- Rule templates 125, 127
- Rules Service 80, 151, 167
 - creating 80
 - properties 167
- Rules view 182
- Runtime Nodes view 136, 138
 - properties and actions 138

S

- sample application 94
- Scheduled ETL Job Service 151
- Scheduled ETL Job service 176
 - properties 176
- Scripting Service 150, 156
 - properties 157
- Security Policies
 - properties and actions 117
- Security Policies view 111
- Security Policy 116
- Service Details view 136, 140
 - properties and actions 141
- Service templates 125
- Services
 - configuration section 106
 - Configuration Server Connector 153
 - Data Mart ETL Services
 - Kettle ETL Service 173
 - Data Mart ETL services 173
 - Database Capture Point 161
 - Database service 169
 - Interaction Server Connector Service 154
 - iWD Extended Statistics Service 170
 - Logging Service 152
 - Rules Service 167
 - Scheduled ETL Job Services 176
 - Scripting Service 156
 - Web Service Capture Point 157
 - XML File Capture Point Service 159
- Services configuration section 136
- Services view 136, 139
 - properties and actions 140
- Setup Utility
 - See iWD Setup Utility
- Solution Details view 136, 137
 - properties and actions 137
- Solutions and Services 136
- square brackets 319
- Standard Genesys to IWD adapter business
 - process 276
- Stat Server 78

Stat Server configuration options 78
 Statistics Adapter Service 151
 String 226

T

Task Action messages 226
 task attribute mapping 245
 task life cycle 24
 Task Notification messages 227
 TaskAssigned message 239
 TaskCanceled message 243
 TaskCompleted message 239
 TaskCreated message 230
 TaskDistributedQueue message 238
 TaskErrorHeld message 241
 TaskHeld message 240
 TaskInfo message 231
 TaskRejected message 244
 TaskRestarted message 242
 TaskResumed message 241
 Tasks
 distribution 20
 prioritization 19
 reprioritization 19
 TaskUpdated message 237
 Tenant Selection 105
 tenant-level configuration 111
 Tomcat 41
 type styles
 conventions 318
 italic 318
 monospace 319
 typographical styles 318

U

Unsaved Changes dialog box 109
 UpdateTask message 236
 updating
 Interaction Server databases 85
 URS 41
 User 101
 User Access 205
 User Profile 110
 user Stat Server option 79

V

verbose Stat Server option 79
 version numbering, document 318

W

Web Service Capture Adapter 17
 Web Service Capture Point 157
 properties 158
 Web Service Capture Point Service 151
 WebSphere 41
 WebSphere MQ Capture Point Service 151
 WFM 13
 WFM Connector Service 151
 Workforce Management 13
 Workforce Management Connector Service 180
 properties 181
 workitem 71

X

XML Capture Adapter 17
 XML File Capture Point Service 151, 159
 properties 160

