Genesys®

intelligent Workload Distribution 8.1

Deployment Guide

About Genesys

Genesys is the world's leading provider of customer service and contact center 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.genesys.com for more information.

Each product has its own documentation for online viewing at the Genesys Documentation 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

Technical Support from VARs

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

Customer Care from Genesys

If you have purchased support directly from Genesys, please contact Genesys Customer Care. Before contacting Customer Care, please refer to the ***Genesys Care Support Guide for On-Premises*** for complete contact information and procedures.

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.genesys.com

Document Version: 81iwd_dep_05-2015_v8.1.102.00

Table of Contents

Chapter 4

Chapter 5

List of Procedures

Preface

Welcome to the *intelligent Workload Distribution 8.1 Deployment Guide.* This document describes how to install and configure intelligent Workload Distribution (iWD).

This document is valid for 8.1.x releases of this product.

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

This preface contains the following sections:

For information about related resources and about the conventions that are used in this document, see the supplementary material starting on page 377.

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.

Starting with iWD 8.1, business rules are authored and deployed in the Genesys Rules Authoring Tool, which is part of the Genesys Rules System. The Genesys Rules System is a set of components that provide business rules functionality for iWD and other Genesys solutions.

Support for eServices and Third Party Media Servers

The iWD 8.1 components work together with Interaction Server and the Genesys Rules System 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 8.1 uses the Interaction Server database to store task information, whereas the 7.6.1 release 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 CD, such as iWD Runtime Node 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 8.1 can be used with Genesys eServices solutions (for example, Genesys E-mail, Genesys Chat, Genesys SMS, and Genesys Social Engagement) 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, SMS, and social media 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 28, and Appendix C, "iWD Business Process (IWDBP)," on page 313.

Integration with Genesys Rules System

The Genesys Rules System is a set of components that provides business rules functionality for use with the iWD solution and other Genesys solutions. It consists of three software components:

- Genesys Rules Development Tool
- Genesys Rules Authoring Tool
- Genesys Rules Engine

Business rule templates are created in the Genesys Rules Development Tool and are published to a rules repository. Users can then incorporate business rule templates into a rule package, by using the Genesys Rules Authoring Tool. Users create and modify rules within a rule package and deploy the rule package to the Genesys Rules Engine. At that point, client applications, such as the iWD business process (IWDBP), can make requests to the Genesys Rules Engine to have rules in the rule package evaluated at various decision points in a task's lifecycle.

Support for Workforce Management

The iWD Standard Rules Template enables you to specify the WFM Activity or Multi-Site Activity to assign to a task as part of a business rule.

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 an attached data element for the interaction that has the name of

the WFM Activity or Multi-Site Activity. Then, in the routing strategy, you can use that attached data element 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 200 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@genesys.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 Customer Care 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 Customer Care

If you have purchased support directly from Genesys, please contact Genesys Customer Care.

Before contacting Customer Care, please refer to the *Genesys Care Support Guide for On-Premises* for complete contact information and procedures.

Document History

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

New in Document Version 8.1.101.00

The document has been updated to support intelligent Workload Distribution release 8.1.1. The following topics have been added or changed since the previous release of this document:

- “Interaction Server Installation” on page 61.
- “iWD Setup Utility” on page 102.
- “Rule Templates” on page 211.
- “Task Management” on page 249.
- “Configuration of List Objects” on page 314.
- “iWD Business Process” on page 317.

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

The intelligent Workload Distribution (iWD) solution 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 captures 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 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 (also referred to as capture points).

The out-of-the-box Capture Adapters are:

- Web Service—To use with source systems that have service-oriented interfaces.

- XML—To integrate with source systems that include the ability to generate XML files.
- Database—To use with systems that do not offer service-oriented interfaces or provide XML output capabilities.

Note: In addition to these out-of-the-box capture adapters, there is an optional Java Message Service (JMS) capture point available that can be used as an add-on to capture tasks from enterprise source systems that support the JMS enterprise messaging system. JMS is a popular messaging infrastructure that is supported by many enterprise applications and middleware components, including SAP NetWeaver, TIBCO, Oracle Fusion, and IBM WebSphere MQ Server.

Interaction Server is already a required component of the iWD solution. Therefore, no additional installation is required to enable the capture point functionality, only some additional configuration. For additional information about capture point functionality within Interaction Server, see the *eServices 8.1 Reference Guide* and *eServices 8.1 User's Guide.*

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.

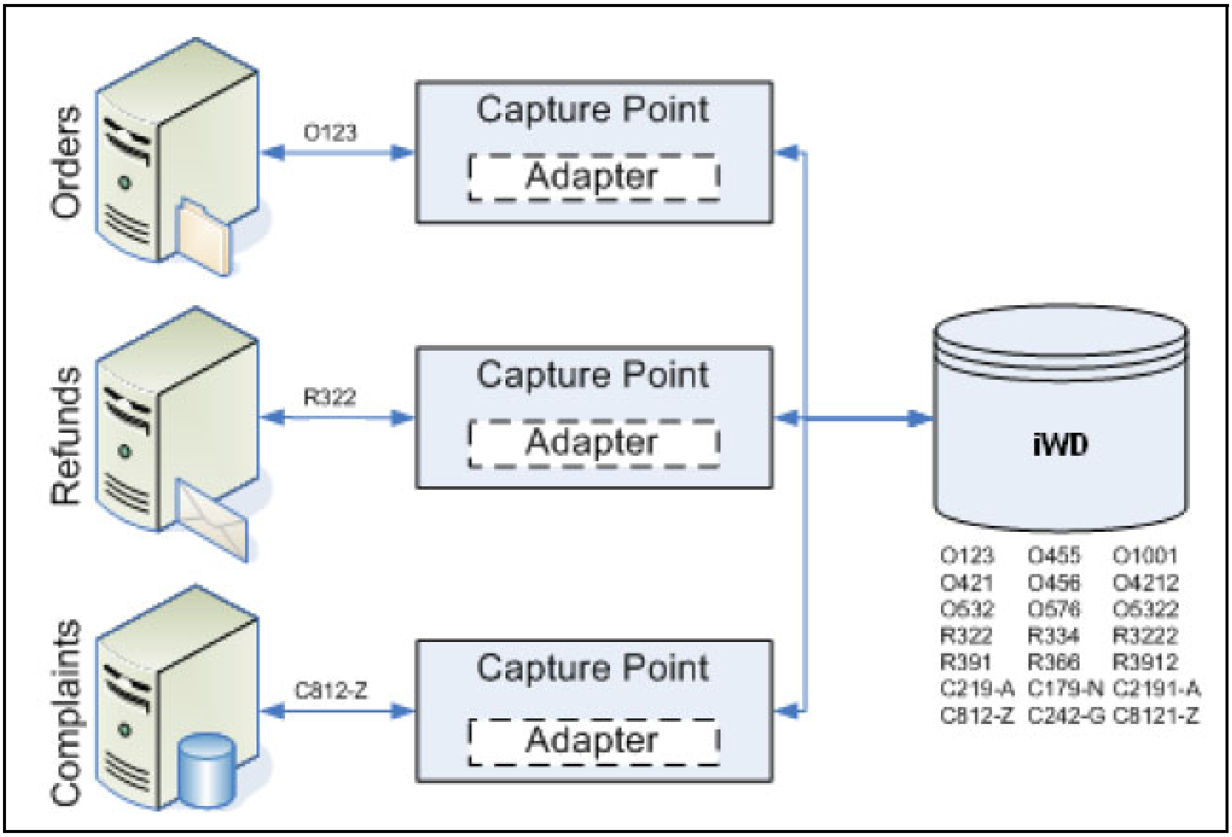


Figure 1: Capture Point Process Diagram

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 Department or 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. These templates, which are the foundation for the business rules that govern iWD, are normally created or modified by IT personnel by using the Genesys Rules Development Tool (GRDT). After the rule templates are published, business users can create or modify rules by using the Genesys Rules Authoring Tool (GRAT), without having to involve IT personnel.

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 employee.

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.

Task Distribution

All iWD tasks are managed through the Genesys Interaction Server and are assigned (routed) to employees by the Genesys Universal Routing Server (URS).

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 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
- Classification and Reprioritization Rules Applied—All 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 employee for processing
- Transfer/Transfer to Queue—Whether the employee transferred the task to another employee 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: Customers can build reports by querying the iWD Data Mart by using the reporting tool of their choice. In addition, Genesys offers the Interactive Insights for iWD product, which includes a data universe and nine out-of-the-box reports, built to be used with the iWD Data Mart.

In addition to the iWD Data Mart, iWD 8.1 interoperates with Interaction Concentrator and Genesys Info Mart for historical reporting. Certain Info Mart 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 (sometimes referred to as Third Party Media), so that these tables in Genesys Info Mart are populated with iWD data.

Refer to the Interaction Concentrator and Genesys Info Mart documentation for more information.

Task Archiving

Task archiving has changed significantly from iWD 8.0/8.1.0 to 8.1.1.

Releases Between 8.0 and 8.1.1

In post-8.0 releases, every task was updated by business rules to provide an *archive destination* and an *expiration timestamp* (`iWD_expirationDateTime`).

When a task reached one of three queues in the iWD Business Process (`iWD_Completed, iWD_Canceled`, or `iWD_Rejected`) and the task's *expiration timestamp* had passed, the task would be moved to an Archive routing strategy.

In that strategy, depending on the value of the *archive destination* attribute associated with the task, the task would either be:

- Deleted
- Moved to another Interaction Server; or;
- The expiration of the task would be rescheduled.

Based on usage patterns of iWD customers, the archiving process has been dramatically simplified in iWD 8.1.1. The notion of archiving a task is really about providing a way for a business user, through the Global Task List, to view tasks that are not currently in process. Moreover, these 'archived' tasks may optionally be maintained in a separate database partition in Interaction Server, to improve system performance.

Release 8.1.1 Starting with iWD 8.1.1, the following changes have been made to task archiving:

- Any task in `iWD_Completed, iWD_Canceled`, or `iWD_Rejected` is now considered archived, as distinct from tasks in any other queues, which are considered current. An `Archived` filter has been provided for the Global Task List, that will display all tasks in any of those three queues.
- When a task is in one of those three queues and its expiration date/time is in the past, the task is moved to a routing strategy where it will be removed from the Interaction Server database. This strategy has been renamed from `Archive` to `Removal` to more accurately describe its function.

  The `Removal` strategy no longer invokes any business rules at the archiving phase, and it no longer provides the ability to move a task to a separate Interaction Server or reschedule the expiration of the task.

  Instead, the logic in the strategy simply deletes (stops processing) the interaction. This will remove the interaction from Interaction Server's `Interactions` table but will maintain the associated events in the Interaction Server Event Log database tables, which is necessary for iWD Data Mart.
- The rule action called `Archive destination "{archive}"` is no longer needed, because expired tasks will always be deleted by the logic in the out-of-box IWDBP Business Process.

The *archiving* rule phase is still provided in the iWD Standard Rules Template, as is the rule action `Archive destination "{archive}"`, for migration purposes. Customers could choose to customize the `Removal` routing strategy if they had a business reason to do some special processing on tasks that have reached their expiration date, other than simply deleting them.

Centralized Logging

iWD 8.1 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 120 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. For more information, see the *Genesys Licensing Guide*.

intelligent Workload Distribution Architecture

The iWD Solution is made up of the following major building blocks:

- iWD Manager—A Java web application that runs on a web application server.
- iWD Runtime Node—A collection of Java services that runs on a web application server.
- Genesys Rules System—A set of software components that are used for rule template development, rules authoring, and rule evaluation. The Rules Engine and Rules Authoring Tool are web applications that run on a web application server. The Rule Development Tool is an Eclipse plug-in.

- Genesys CIM Platform—Core Genesys components that provide interaction management (Interaction Server), routing (Universal Routing Server), employee presence and employee and queue-based real-time statistics (Stat Server), and configuration and management services (Genesys Management Framework, including Configuration Server and Message Server).

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

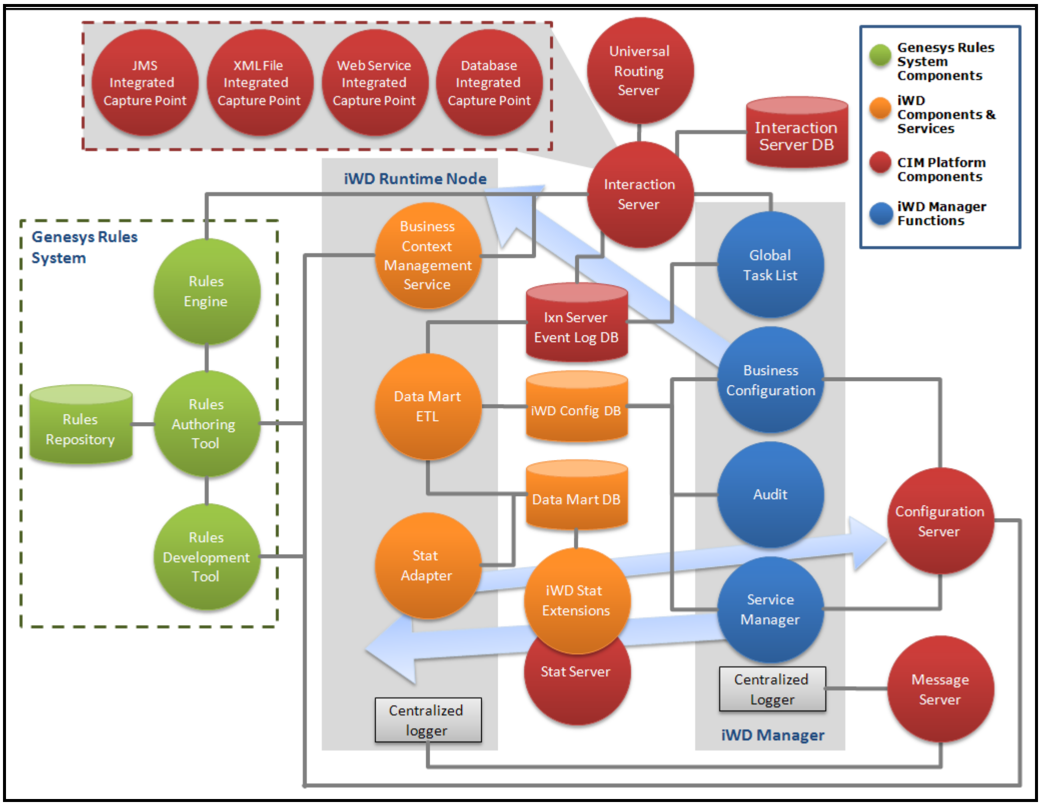
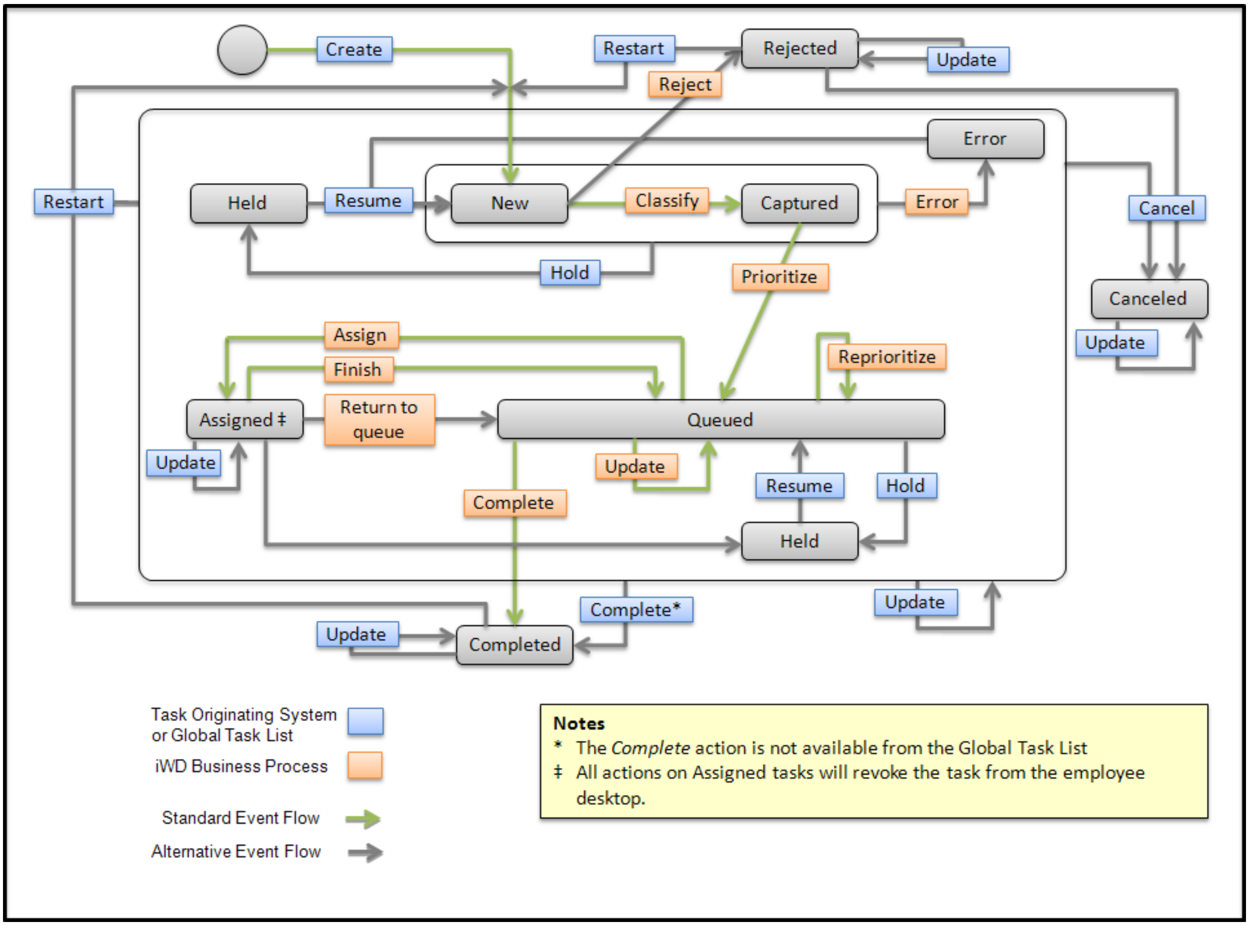


Figure 2: intelligent Workload Distribution High-Level Architecture

Task Life Cycle

Figure 3 illustrates the overall task life cycle in iWD:



Notes
* The *Complete* action is not available from the Global Task List
‡ All actions on Assigned tasks will revoke the task from the employee desktop.

Figure 3: Task Life Cycle

New in This Release

This section lists new features and functionality for all intelligent Workload Distribution 8.1 releases.

Release 8.1.1 The following new features and functionality are introduced in this release:

- The iWD Business Process (IWDBP) has been updated to support interaction with the Genesys Universal Contact Server (UCS). This update allows the business process logic to update the UCS database to mark an interaction as *done* (the value of the `Status` column in the `Interaction` table in the UCS database is set to `3`), even when it will persist for some time in the Interaction Server database.

- The `Archive` routing strategy and some associated business logic within IWDBP has been greatly simplified, to reflect best practices for archiving iWD interactions.
- iWD Manager has been enhanced to allow a user of the Global Task List to cancel, restart, or update a `Held` task. Additionally, `Assigned` tasks can be canceled, held, restarted or updated by a user through the Global Task List.
- Several new user authentication features have been added:
  - The ability for iWD Manager users to change their password on demand.
  - Display within iWD Manager of the last successful login date and time for the user.
  - Display to the user of any meta-text provided by a RADIUS server, when Genesys Configuration Server is configured for external authentication with the RADIUS server.
  - Ability for the iWD Manager users to change their password on the iWD Manager login screen, when forced to do so by a System Administrator.
- Support for:
  - Microsoft Internet Explorer 9, in compatibility mode. No special configuration is required.
  - Websphere Application Server 8.x
  - Apache Tomcat 7
  - Mozilla Firefox version 17

The following new features and functionality are introduced in this release:

- iWD integration with Genesys Rules System (GRS)—Consists of three software components: Genesys Rules Development Tool, Genesys Rules Authoring Tool, and Genesys Rules Engine. See also, “Integration with Genesys Rules System” on page 15.

  This integration between iWD and GRS led to the following changes:
  - Business rule templates are now created and edited in the Genesys Rules Development Tool, an Eclipse-based GUI. They are no longer created in iWD Manager.
  - Business rules are now created and edited through the Genesys Rules Authoring Tool, which can be launched from iWD Manager through a single-sign on mechanism. They are no longer authored within iWD Manager.
  - Any change made to the business structure (Solution, Department, and/or Process) that is created and modified through iWD Manager must now be *pushed* to the Genesys Rules System. This is done through an option in iWD Manager.

    - The iWD business process (IWDBP) is now updated to integrate with the Genesys Rules Engine.
- The Global Task List in iWD Manager is enhanced in several ways, including:
    - Seven new (in addition to the existing) filter criteria can be used to build custom filters.
    - A custom attribute can be added as a column in the task list, when a custom filter is created for the Global Task List.
    - User friendly on-screen labels for any iWD custom attribute can now be created to enhance the way in which attributes appear on the Global Task List. This is done within a resource file that resides under the iWD Manager web application on the customer's application server.
    - The Global Task List can now display an icon for each different interaction (task) media type. Out-of-the-box icons are now provided for many media types and new icons can be uploaded into iWD Manager for custom media types, or to replace any of the out of the box icons that are provided by iWD.
    - Hovering a mouse over the task list, now enables you to see the full text of any attribute in the task list as a tooltip.
- Support for the Genesys Social Engagement solution is now enhanced. When Genesys Configuration Server Tenant's properties include the section `iwd.profile` with the `social-messaging` option set to `true`, iWD Manager automatically creates the configuration objects and executes the database updates that are necessary for social media interactions to be properly displayed in the iWD Global Task List.
- There is a new type of service template called Generic Capture Point, that is used whenever an Interaction Server Integrated Capture Point is added to an iWD Solution. The Generic Capture Point establishes a `Capture Point ID` and `Capture Point Name` for the Integrated Capture Point, ensuring that the interactions (tasks) that are captured through that Integrated Capture Point are visible in the iWD Global Task List, can be referenced within iWD business rules, and tracked properly within the iWD Data Mart.
- Changes in platform support, including discontinued support. See the *Genesys Supported Operating Environment Reference Guide.*

iWD Business Process (IWDBP)

iWD 8.1 is packaged with an out-of-the-box Business Process. The Business Process is installed by using the iWD Setup Utility and is also provided with the iWD Manager installation so it can be manually imported through the Genesys Interaction Routing Designer. It is made up of a set of Interaction Queues that map to the iWD state model (`NEW`, `ERRORHELD`, `CAPTURED`, `COMPLETED`, `CANCELED`, `REJECTED`, and `QUEUED`).

Within this Business Process, from within a routing strategy, External Service Protocol (ESP) blocks are used to invoke methods of the Business Context Management Service (BCMS) and Genesys Rules Engine (GRE). 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 communicates with Interaction Server 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. If the Genesys iWD Business Process is not used, iWD Data Mart and iWD Global Task List functionality may be limited.

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 BCMS and ESP block to invoke the Genesys Rules Engine.

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 C on page 313.

Chapter

2 iWD Best Practices

This chapter provides some best practice guidelines to use when you are planning, deploying, and configuring intelligent Workload Distribution (iWD). It contains the following sections:

Design Phase

This section includes information about best practices to consider in the design phase of your iWD implementation.

Review Messaging Between the Source System and iWD

It is important to understand how the source system works. In particular, which triggers on the source system will be used to generate iWD messages (such as, `CreateTask` or `UpdateTask`). The involvement of all stakeholders—business analysts and technical personnel, representing both the source system and the iWD business process—is key to a successful outcome.

Map Out Complete Task Flow or State Model

During the design phase, it is important to document the end-to-end message flows from the source system all the way to the agent desktop or toolbar, that describe messaging for each scenario that might be encountered. Ladder diagrams that show the various actors in the system can be very useful for this exercise.

Also, ask the question, "When is a task considered completed from a business perspective?" The answer affects both reporting and distribution. For example:

- A task enters `iWD_Queued` and is distributed to and handled by an employee and then, is passed on (for example, it is completed or transferred by the employee or deferred for later completion). In which of these scenarios do you want the task to be considered completed from the iWD state model perspective? Completed in this sense means completed by the Department and Process.
- Do you want the task to be completed only when it is completed within the source system? This can allow for true end-to-end reporting that captures when the task is created in the source system until the time it is completed in the source system, plus the time spent on the task by any iWD-enabled employees.

This discussion becomes particularly relevant when you design the triggers on the source system for the iWD operations such as `CreateTask`, `UpdateTask`, and `RestartTask`.

Don't Forget About the Desktop

There might be times when task-related logic needs to be implemented at the employee desktop, or toolbar. For example, if an `UpdateTask` message is received while the task is assigned to an employee, the desktop might need to react in different ways, depending on the content of the update message. In those cases, it is important to include the desktop or toolbar developer during the design phase when various scenarios are discussed and documented.

Task Capture

This section includes information about best practices to consider when you are planning and configuring task capture for your iWD solution.

Selecting the Appropriate Capture Point

The capture point you select will depend on the capabilities of the source system from which you are capturing tasks. In some cases, there will be multiple options you will be selecting from. Therefore, it is useful to know the capabilities and limitations of the various capture points.

When possible, you should use the JMS Integrated Capture Point if the JMS (Java Message Service) enterprise messaging service is supported by your source system. This capture point is bi-directional, supporting an input queue and an output queue. Since it uses an enterprise messaging system, it is more reliable than other types of communication. For example, if iWD has a notification to provide to the source system, that notification will be placed in a JMS message queue by the JMS Integrated Capture Point. If there is a loss of communication between the JMS Provider and the source system, the notification message will remain in queue until communication is restored.

The only other bi-directional capture point is the Database Capture Point, which is very flexible. However, to leverage the bi-directionality, it is necessary to update tables on the source system database. In some environments, this will not be possible.

The Web Service Integrated Capture Point is an excellent choice for integration with any source system that has the capability of invoking SOAP/HTTP messages from within its workflow. The message set of the Web Service Integrated Capture Point is very straightforward and is easy to integrate. The SOAP payload, in XML format, is easy to understand and is logically organized.

While the Web Service Integrated Capture Point is not bi-directional, you can use the Web Service Capture Point from the source system to request the latest status for a particular task, including the current values of all the task attributes. This could be done from the source system just before taking certain actions on the source system, to ensure that the latest updates that might have occurred on the Genesys side, can be propagated to the source system.

The XML file capture point is a good option when it is being integrated with legacy host systems that do not have support for web services or modern messaging systems. In most cases, these systems can generate flat files that consist of lists of attributes for each work item in its database. These flat files could be converted to an XML format and then read by the iWD XML File Integrated Capture Point.

Proper Use of Extended Attributes

Where possible, use iWD extended task attributes to map the source system's data fields that you intend to use in business rules. This is preferable to the use of custom attributes, because the extended attributes are out-of-the-box. That is, no additional configuration is required to use them. Columns are already included in the Interaction Server databases for these attributes and their associated `Interaction Custom Properties` are defined in Configuration Server. In addition, they are better supported in iWD Data Mart because some of these extended attributes are already dimensions in the iWD Data Mart database and can be queried out-of-the-box.

Optimizing the Use of Custom Attributes

Do not capture unnecessary custom attributes from the source system. This increases the size of the attached data and the Interaction Server and Event Log databases. Include a custom attribute only within the `CreateTask` message, if you plan to use the attribute in one or more of the following ways:

- In business rules
- In a routing strategy
- To display to a user through the Global Task List
- To display to a user through an *agent desktop,* such as Genesys Interaction Workspace
- To use for reporting

To display data to an employee on the desktop, consider the desktop on which the employee will access work on the task. If the employee is working directly on the source system desktop to fulfill the task, then it might not be necessary to capture attributes, if only to provide this data to the employee.

Business Rules Configuration

This section includes information about best practices to consider when you are configuring business rules for your iWD solution.

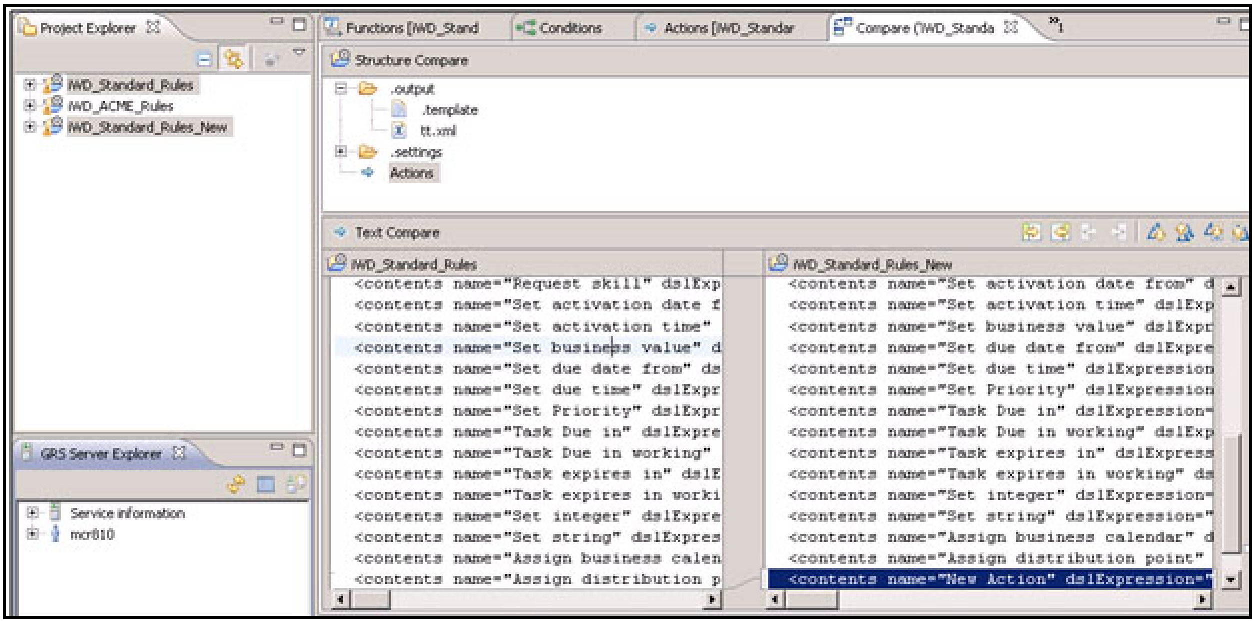
Use Custom Rule Templates

It is tempting to simply augment the iWD Standard Rules Template to meet your specific business requirements. However, it is a best practice to create one or more custom rule templates to add new rule conditions and actions that you require to meet your business requirements. There are several a few advantages to this approach, such as:

1. Genesys might release new versions of the iWD Standard Rules Template from time-to-time. Importing a new version of the Standard Rules Template into the Genesys Rules Development Tool requires that you delete (or rename) the existing version. Therefore, any custom rule conditions or actions that you added to the Standard Rules Template would be lost.
2. By keeping the iWD Standard Rules Template intact, it allows you to associate it with the `Environment` tenant in Configuration Server. In a multi-tenant environment, this enables a common, standard set of rule conditions and actions that can be accessed by all tenants.
3. Access control can be applied to rule templates, because each template is represented by a separate `Script` object in Configuration Server. Therefore, multiple rule templates can be created, segregating different types of rule conditions and actions that will be accessed by different types of users or by different functional areas of the business. Normally, you will want the Standard Rules Template to be accessible by all users whereas, you might have other templates where basic rule conditions and actions are in one template and more advanced conditions and actions in a separate

template. You can then use access control on the associated `Script` objects to determine which users will be able to access specific templates through the Genesys Rules Authoring Tool.

If you do modify the iWD Standard Rules Template, Eclipse provides a way to compare your modified template project with the original version that is included in the Genesys iWD Manager Installation Package. Just rename your modified version of the template project, and then import the original Genesys version. Select both projects in the `Project Explorer`, right-click, and choose `Compare With Each Other`. A `Compare` view will display the differences in the `Actions`, `Conditions`, `Functions`, and `Parameters`. In the example in Figure 4, one new `Action` has been added to the `iWD_Standard_Rules_New` project.



```
<contents name="Request skill" dslExp
<contents name="Set activation date f
<contents name="Set activation time"
<contents name="Set business value" d
<contents name="Set due date from" ds
<contents name="Set due time" dslExpr
<contents name="Set Priority" dslExpr
<contents name="Task Due in" dslExpre
<contents name="Task Due in working"
<contents name="Task expires in" dslE
<contents name="Task expires in worki
<contents name="Set integer" dslExpre
<contents name="Set string" dslExpres
<contents name="Assign business calen
<contents name="Assign distribution p
```

```
<contents name="Set activation date from" d
<contents name="Set activation time" dslExp
<contents name="Set business value" dslExpr
<contents name="Set due date from" dslExpre
<contents name="Set due time" dslExpression
<contents name="Set Priority" dslExpression
<contents name="Task Due in" dslExpression=
<contents name="Task Due in working" dslExp
<contents name="Task expires in" dslExpress
<contents name="Task expires in working" ds
<contents name="Set integer" dslExpression=
<contents name="Set string" dslExpression="
<contents name="Assign business calendar" d
<contents name="Assign distribution point"
<contents name="New Action" dslExpression="
```

Figure 4: Compare Modified Template with Original (Genesys) Version

Design of Rules Hierarchy

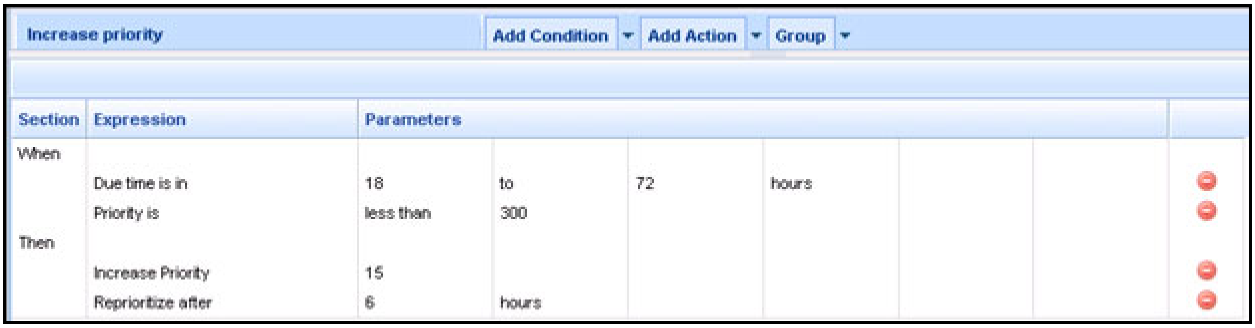
It is useful to create business rules at different levels of the business structure (for example, Global Rules, Department-level, Process-level), rather than putting all rules at the Global level. Not only does this configuration make troubleshooting simpler, it also enables you to provide access control to specific sets of rules. Moreover, it enables you to set default rules. For example, at the Global Rules level you might set a default priority or a default due date for all tasks that meet specific criteria. You can then override those defaults at a lower level of the business hierarchy, based on various conditions.

Use Prioritization Ranges

It can be very useful to define priority ranges for different types of tasks. Priority ranges define the minimum and maximum priorities that can be assigned to any type of tasks.These tasks are enforced when the tasks are assigned their initial priorities and when they are reprioritized over time.

For example, in a blended environment you might reserve priorities 501+ for voice calls, whereas 401-500 are for the highest-value off-phone tasks, 301-400 are for the next highest-value, and so on.

Through business rules, you can ensure that a particular type of task's priority never exceeds an upper ceiling. You can do this in your prioritization or reprioritization rule by adding a rule condition, such as `Priority is {operator} {priority}`, where `{operator} = less than`. See an example in Figure 5.



Section	Expression	Parameters						
When								
	Due time is in	18	to	72	hours			⊖
	Priority is	less than	300					⊖
Then								
	Increase Priority	15						⊖
	Reprioritize after	6	hours					⊖

Figure 5: Upper Limit on Rule Priority

By maintaining tasks within predefined priority ranges, it is easier to troubleshoot anything that might happen with these tasks, and easier to predict what will actually be routed to an employee when they become available for work.

Do Not Reprioritize Interactions Frequently

It is important to reprioritize tasks at reasonable time intervals. That is, if a task is not due for 3 days, and your business normally operates with a significant task backlog, it does not make sense to reprioritize that task every 15 minutes. That will just consume unnecessary resources that can put a stress on the system, when there are tens of thousands, or hundreds of thousands of tasks in queue. Therefore, plan your reprioritization intervals so that tasks that are not due for several days are only reprioritized once, or a few times per day, versus tasks that are due within the current working day (or a shorter time frame). Those tasks could be reprioritized once per hour.

The appropriate reprioritization intervals should be based on an analysis of your backlog and how soon you expect to work through it. Remember that you can set up different reprioritization intervals, based not only on when a task is due, but on any other criteria as well. The criteria might include department, process, business value, current priority, any custom attribute, or combination of custom attributes.

It can be an extremely helpful exercise to graph out the different types of tasks and the way in which you expect the reprioritization of the tasks to occur over time. You can start by putting this data into an Excel spreadsheet, and then, within Excel, automatically generate a line chart. Put each task in a different color. The more intersecting lines you have on your graph, the more confusion you can expect to have when you put the system into operation.

Configuration

This section includes information about best practices to consider when you are configuring your iWD solution.

Use Event Log Filtering

You can configure which events will be stored in the Interaction Server Event Log database. Doing so can reduce the number of rows in that database for *active* tasks (tasks that have not yet expired and therefore, have not been purged by the `iWD ETL Prune` job). The events that are necessary for iWD Data Mart and the task's Event History to work properly are shown in Table 1 on page 37.

Table 1: Interaction Server Events Required for Proper Functioning of iWD

Event identifier	Event name
104	`EventPropertiesChanged`
132	`EventPartyAdded`
133	`EventPartyRemoved`
161	`EventProcessingStopped`
162	`EventPlacedInQueue`
163	`EventPlacedInWorkbin`
193	`EventExternalServiceRequested`
194	`EventExternalServiceResponded`
200	`EventHeld`
201	`EventResume`

You can filter out all other events by configuring the Interaction Server Event Log Database Access Point application option `event-filter-by-id`. This option is configured in the `event-filtering` section. The value of the `event-filter-by-id` option will be a comma-separated list of the events you want to be logged. All other events will not be logged.

The list of events shown in Table 1 is appropriate if you are using the out-of-the-box iWD business process (IWDBP). The best way to analyze this is to turn off the `event-filter-by-id` option temporarily and to put one task through the entire business process, including reprioritization, handling at an end user desktop, any re-queuing, and so on. Then, you can review the full list of events that have been captured by doing a query on the Event Log Database table, for a particular Interaction ID. If you do not require any custom reporting and have no need for the events other than those listed in the table above, you can apply the filter, as described.

Align Business Structure and Business Requirements

In some businesses, the way you define Departments and Processes in iWD will directly align with how the business views distribution and reporting.

In other cases, consider aligning Departments and Processes with your reporting requirements and use Genesys skills to align with distribution. This is the recommended approach because the Departments and Processes can then be used as input in the Data Mart plug-ins—that is, the pre-defined attributes of Department and Process can be used to support the reporting metrics and dimensions. This makes it easier to provide statistics from a business point of view.

Consider Using Multiple iWD Tenants

Consider configuring more than one iWD managed tenant, where each tenant aligns to a different business unit. This allows you to configure dedicated custom attributes in iWD Data Mart for each business unit. It also reduces the amount of data iWD Data Mart has to process from the Interaction Server Event Log database. This means you will need to set up multiple iWD Data Mart instances, but this configuration is more scalable.

Note: Using tenants is a recommended solution, but it is possible to use different solutions, because iWD Data Mart is a per-solution entity.

Load Balance BCMS and GRE in High Volume Deployments

If your iWD solution has particularly high volumes or uses frequent reprioritization, it might be useful to set up multiple Business Context Management Services (BCMS) (see “Configuring Multiple Business Context Management Services” on page 266) and multiple Genesys Rules Engines (GRE) in a load-balanced configuration. Consider updating the out-of-the-box IWDBP business process to add a subroutine that can handle BCMS load balancing and retries, with multiple runtime nodes within the solution. You can make the number of retry attempts configurable as a strategy variable or within a List Object so the value can be modified without changing the strategy itself.

Multiple Application Server Instances

It might be beneficial to set up the iWD Manager, iWD Runtime Node, and iWD Data Mart web applications on dedicated application server instances. In fact, Genesys recommends that you always use a dedicated application server instance for iWD Data Mart. However, having the other applications on dedicated instances can also improve scalability and availability.

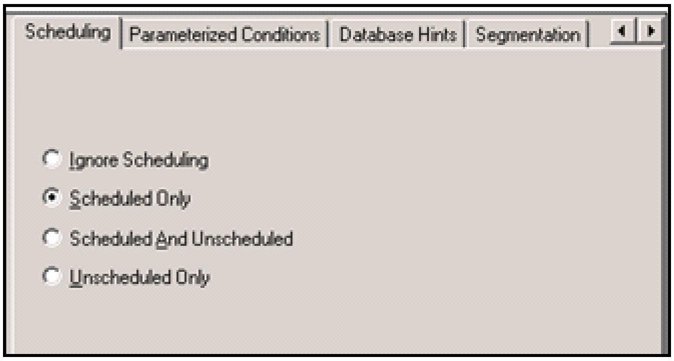
Task Distribution and Routing

This section includes information about best practices to consider when you are planning and configuring the task distribution and routing components of your iWD solution.

Use Scheduling in the Queue Views

Queue views define the criteria that must be met for a task to be submitted from a queue to a Distribution routing strategy. Using the `Scheduling` tab can prevent a task from bouncing between Interaction Server and Universal Routing Server (URS), especially when there are no agents logged in to handle tasks. In this case, the `ScheduledAt` attribute can be used to reschedule submission of tasks back to URS.

Note: For information about the `ScheduledAt` property, see the topic "Setting the ScheduledAt Property" in the *Universal Routing 8.1 Business Process User's Guide.*

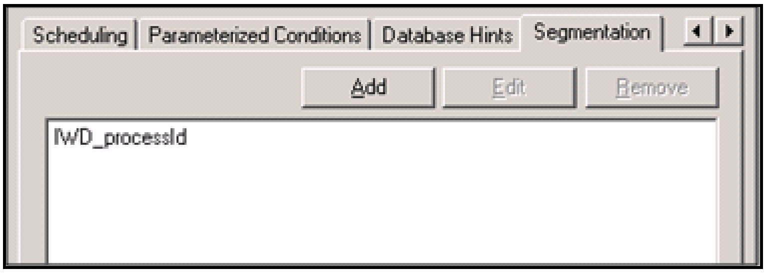


○ Ignore Scheduling
● Scheduled Only
○ Scheduled And Unscheduled
○ Unscheduled Only

Figure 6: Scheduling Tab—URS

Use Segmented Views to Keep Agents Busy

Consider using segmentation on the `Queue` view from the `iWD_Queued` queue (or custom interaction queue in your iWD business process) to the `Distribution` routing strategy. Segmenting interactions, based on iWD Process (`IWD_processId`), or by skill might be a good idea. This ensures that all types of work (tasks) will be submitted for distribution even if these tasks (in a specific process) have low priorities. There might be specific agents that are dedicated to manage or handle these low priority tasks. However, if they have low priorities, they might never be submitted to a distribution strategy.



IWD_processId

Figure 7: Segmentation Tab

Use Triggers in the Routing Strategy

There are times when a task is updated by the source system; for example, an agent who is not iWD-enabled has modified the task or an external system has modified the task. This task update might impact the classification, prioritization, or distribution of the task. Setting triggers on the `Distribution` strategy to react to specific changes in attached data elements enables you to reclassify, reprioritize, and redistribute tasks, as required.

There is a function in IRD/URS 8.x called `SetUpdateTrigger[],` which enables you to specify an attached data key that will be monitored for dynamic changes while an interaction is sitting in the `Target` block, waiting for an available agent.

Example: Specifying Attached Data Key

- An interaction is waiting for an available agent in a `Target` block in the `Distribution` strategy, with a time out of `30` minutes before it goes to the red port. During this time, it possible that the source system will send an `UpdateTask` message (for example, if another agent pulled the task manually from the source system).
- If the `SetUpdateTrigger[]` function is enabled in the strategy for a specific attached data element, and then, the value is updated, the interaction goes to the red port automatically, enabling you to evaluate the `UpdateTask` message and take control of the interaction in the strategy.
- If `SetUpdateTrigger[]` is not set up in the `Distribution` routing strategy to handle this, then the interaction might be stuck in the Target block and could get distributed to other agents, which could cause a loss of synchronization between the source system and IWD.

A recommended approach is to create a custom attribute such as `iwdAction` in the `SetUpdateTrigger[]` function, with possible values of `CreateTask`, `UpdateTask`, and so on. This attribute would be set by the source system in any `CreateTask` or `UpdateTask` messages that would be generated from the source system. After evaluating the message within the IWDBP business process, its value can be set to `CLEARED`.

Apart from using the `iwdAction` custom attribute in the `SetUpdateTrigger` function, this attribute can also be used in most of the `Interaction Queue` or `Workbin Views` to determine if the task has been updated (by using the `UpdateTask` message) while sitting in those `Queues` or `Workbins`. This can be done by creating a new `View` from these Queues and `Workbins` called, for example, `BackEnd Update`, with this condition:

```
iwdAction !='CLEARED'
```

If this condition is met, the interaction can be submitted to a routing strategy that evaluates the update task message and performs the appropriate actions, such as distributing it to a specific agent, applying rules, or ignoring the update and sending it back its original location.

Avoid Looping in Strategies

It is important to prevent looping within your routing strategies. It can place a lot of load on the Business Context Management Service (BCMS) and/or the Genesys Rules Engine (GRE) if reclassification and reprioritisation are occurring over and over again. If these requests fail because BCMS and/or GRE is down, or a rule is not applied successfully due to a syntax error that cannot be caught during design-time, consider taking one of the following actions:

- Use scheduling to delay the next attempt at rules evaluation.
- Place the task into the `iWD_ErrorHeld` queue to stop a *bad* task from impacting the overall system. For example, you might check the value of the `IWD_processId` attribute to verify that it has been classified correctly—that is, it is not `NULL`. From the `iWD_ErrorHeld` queue you might resubmit the task into an error handling strategy, which then sends the task back to the `iWD_New` queue or performs other integrity checking on the task's attached data.

Consider Pull Versus Push Task Distribution

Many businesses like iWD's push-based model of task distribution. It helps avoid the common problem of employees picking the easiest task to work on next, and to ensure that employees are always working on the highest-priority task.

However, it is likely that some business workflows will require a pull (workbin) model. So consider the role that Agent Group Workbins might play in your solution.

Also consider that agents might need to hold on to a task for a period of time if they cannot complete it immediately. This might require the agent to open and close the task multiple times before finally completing, or otherwise dispositioning, the task. In this case, you will probably want to use personal Agent Workbins. In both cases, you will need to consider the reporting and distribution implications.

Reporting

This section provides information about iWD Reporting functions.

iWD Statistics in CCPulse+

You can display performance statistics that are related to iWD Departments and Processes, through CCPulse+. These statistics are sourced from the iWD Data Mart. No agent performance metrics from the iWD Data Mart are fed to Stat Server and therefore, to CCPulse+. However, you can use the standard Stat Server and CCPulse+ agent metrics such as Average Handle Time, Total Login Time, Total Talk Time, and so on. All of these agent status-related statistics will be applicable to iWD work item handling as well. For Average Handle Time, you can apply filters, including filtering by media type or any other attached data key-value pair such as iWD Department or Process.

Chapter

3 Deployment Overview

This chapter provides an overview of iWD deployment. This chapter contains the following sections:

Installation Overview

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

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

iWD installation consists of two steps:

- “Application Installation” on page 44.
- “Database Preparation” on page 44.

Note: For detailed installation information, see Chapter 4, “Installation,” on page 53.

~~Compatibility with URS and IRD~~

~~iWD 8.1 is compatible with Genesys Universal Routing Server (URS) 7.6 or higher. Interaction Routing Designer (IRD) 8.1.2 or higher is required for the provided business process (IWDBP) to work.~~

Database Preparation

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

- iWD Configuration database—Stores iWD system and business logic configuration such as services and processes (see Chapter 5, “Configuration,” on page 127 for descriptions of these objects).
- Genesys Rules System rule repository—Stores business rule templates and business rules. For more information, see the *Genesys Rules System 8.1 Deployment Guide.*
- Interaction Server databases—Stores iWD tasks and related task events. Make sure that you have installed and configured the Interaction Server and its associated databases, as described in the *eServices* (*Multimedia) 8.1 Deployment Guide*.

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

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.

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 by iWD.

A basic iWD deployment, such as in a lab environment, 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. These deployments provide more controlled resource partitioning and allow load distribution across multiple physical servers. For a production deployment, Genesys recommends that you deploy iWD Data Mart services on a dedicated iWD runtime node.

In addition to the iWD Manager and iWD runtime node installation, Genesys Rules System must be installed. This installation includes the Genesys Rules Development Tool, which is an Eclipse plug-in for business rule template development, and two web applications that run on a Java application server. These web applications are the Genesys Rules Authoring Tool, which is used for authoring and deploying business rules, and the Genesys Rules Engine, which is an engine that receives and processes requests for business rule evaluation from clients, such as the routing strategies in the iWD Business Process (IWDBP).

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 following topics provide an overview of different system configuration aspects:

- “Tenants” on page 45
- “Solutions” on page 46
- “Services” on page 47

For a detailed system configuration reference, see Chapter 5, “Configuration,” on page 127—specifically, the following sections: “General Configuration” on page 141, “Modules and Components” on page 162, and “Solutions and Services” on page 166.

Tenants

iWD configuration supports multi-tenancy. iWD automatically creates a root tenant that is named `System` by default. The `System` tenant is automatically mapped to the Environment tenant in Genesys Configuration Server. 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 maps 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 managed 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 `Data Mart`. The specific functionality is implemented as a collection of components that can be services or business logic templates (such as metrics 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.
  - Metric template.
  - Scripts—These are transformation scripts that are only used with the legacy iWD capture point services. For more information, see Appendix B, “Legacy iWD Capture Point Services,” on page 277.
- 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 47)—iWD runtime application instances that are within the Java application server in which services are being run

- Services (see page 47)—Services that enable iWD functionality, such as Data Mart, Statistics Adapter, and logging.
- Business logic configuration (see page 47)—Primarily the configuration of iWD departments and processes.

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 43.

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 loading data into the Data Mart. Refer to “iWD Services” on page 178 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 and processes. This phase also includes the definition of business rules for use in task processing. 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 48
- “Rules” on page 49
- “Task Classification” on page 50
- “Task Prioritization” on page 51
- “Business Calendars” on page 52

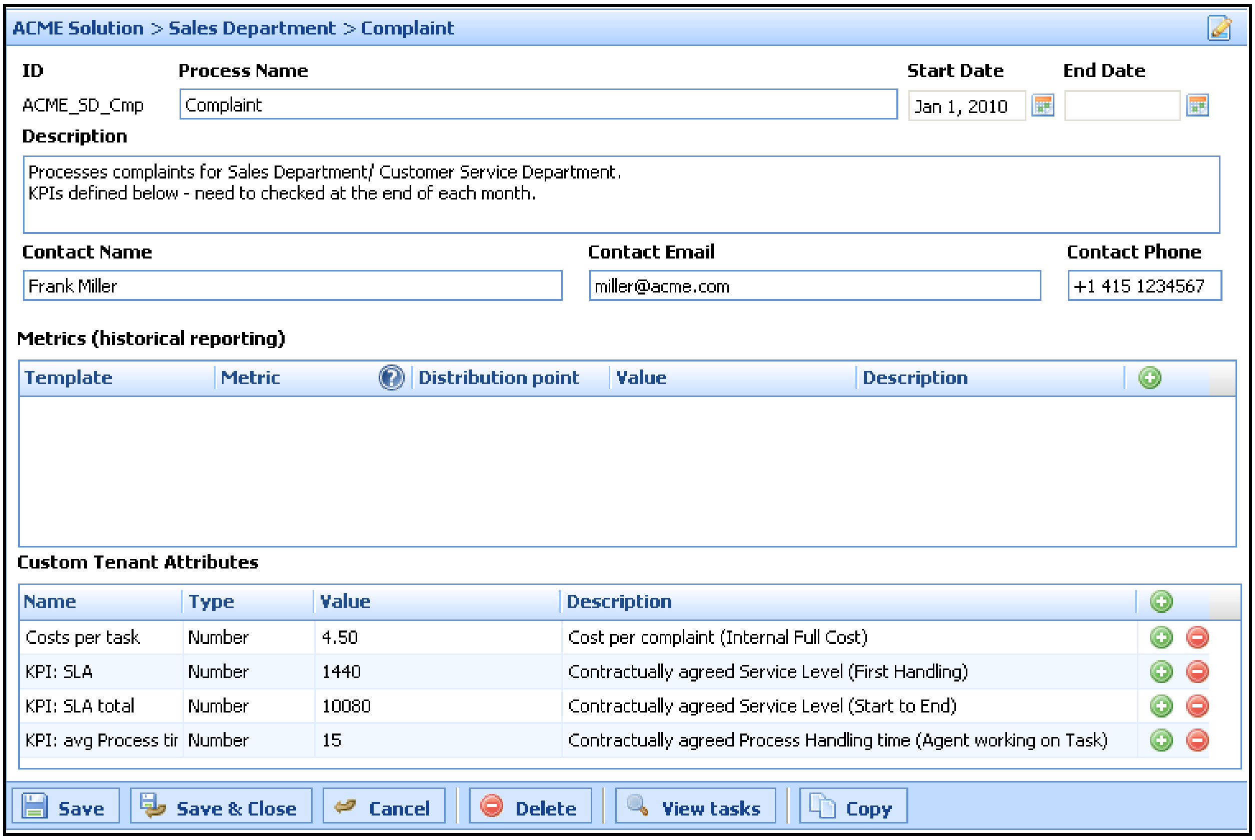
For a detailed business configuration reference, see the section “Departments and Processes” on page 202 of Chapter 5.

Departments and Processes

Departments represent organizational entities 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 50 and “Task Prioritization” on page 51.

Departments and Processes are created in iWD Manager and are then *pushed* (synchronized) to the Genesys Configuration Server so that they become available to the Genesys Rules System. This is required so that users that are creating business rules through the Genesys Rules Authoring Tool (a component of the Genesys Rules System) are able to do so within the business structure that is defined by users in iWD Manager.

Each department and process allows for the definition of any number of custom attributes to a department/process in order to provide additional enterprise-specific context for reporting purposes, but only a maximum combination of 5 can be used in reporting. Figure 8 shows an example of a process.



ACME Solution > Sales Department > Complaint

ID	Process Name	Start Date	End Date
ACME_SD_Cmp	Complaint	Jan 1, 2010	

Description

Processes complaints for Sales Department/ Customer Service Department.
KPIs defined below - need to checked at the end of each month.

Contact Name	Contact Email	Contact Phone
Frank Miller	miller@acme.com	+1 415 1234567

Metrics (historical reporting)

Template	Metric	Distribution point	Value	Description

Custom Tenant Attributes

Name	Type	Value	Description
Costs per task	Number	4.50	Cost per complaint (Internal Full Cost)
KPI: SLA	Number	1440	Contractually agreed Service Level (First Handling)
KPI: SLA total	Number	10080	Contractually agreed Service Level (Start to End)
KPI: avg Process tir	Number	15	Contractually agreed Process Handling time (Agent working on Task)

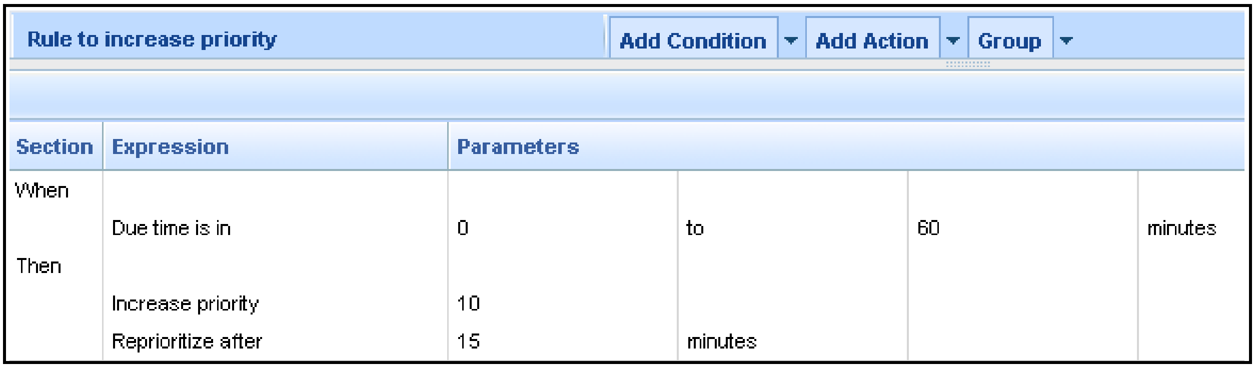
Figure 8: Example Process

Rules

Part of configuring the business logic for departments and processes in iWD is configuring and associating rules. All business rule authoring for iWD is done through the Genesys Rules Authoring Tool, which is a component of the Genesys Rules System. These rules define the task-handling business logic that is applicable to the departments, processes, or the entire system. Generally, a rule is represented by zero, one or more conditions and one or more actions. If there are no conditions for the rule, its actions will always be executed. If all of the conditions are true, all of the actions are executed. If any condition is not true, none of the actions are executed.

Rules are expressed in an easy-to-understand human language, such as, “If the 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.

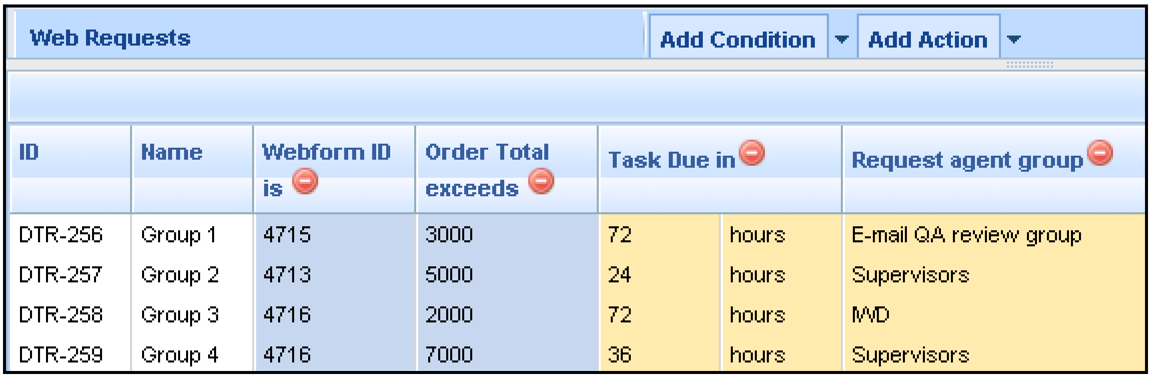
Rules can be defined in one of two ways: as a linear rule or as a decision table. Linear rules are intended for complex rules that have many conditions and/or actions. Each condition or action is represented by a single line in the rule. Figure 9 shows an example of a linear rule.



Section	Expression	Parameters			
When					
	Due time is in	0	to	60	minutes
Then					
	Increase priority	10			
	Reprioritize after	15	minutes		

Figure 9: 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 10 shows an example of a decision table.



ID	Name	Webform ID is	Order Total exceeds	Task Due in		Request agent group
DTR-256	Group 1	4715	3000	72	hours	E-mail QA review group
DTR-257	Group 2	4713	5000	24	hours	Supervisors
DTR-258	Group 3	4716	2000	72	hours	IWD
DTR-259	Group 4	4716	7000	36	hours	Supervisors

Figure 10: Decision Table Example

Task Classification

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

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

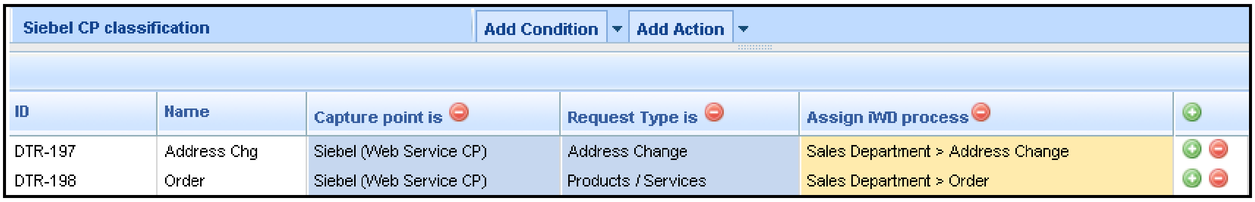
- Package-level rules (also known as Global Rules)
- Department
- Process

If rules are defined for more than one context, they are evaluated in sequence, as previously listed. After a process has been assigned to a task, additional classification rules are evaluated that have been defined, first at the rule package or global level, then at the department level, and finally at the process level. Figure 11 shows an example of Task Classification.

A process must be assigned during the classification phase. It can be assigned in two places—at the global level, or at the department level. (A department can be assigned at the global level and then, the rule evaluation can continue at the department level to actually select a process.)

Important Information To Note:

In iWD 7.6.1 and 8.0, you could also define classification rules at the Capture Point level. In iWD 8.1, with the introduction of the Genesys Rules System, Capture Point rules are now configured at the rule package (global) level, by using the rule condition `Capture Point is`. See Figure 11.



ID	Name	Capture point is	Request Type is	Assign iWD process	⊕
DTR-197	Address Chg	Siebel (Web Service CP)	Address Change	Sales Department > Address Change	⊕ ⊖
DTR-198	Order	Siebel (Web Service CP)	Products / Services	Sales Department > Order	⊕ ⊖

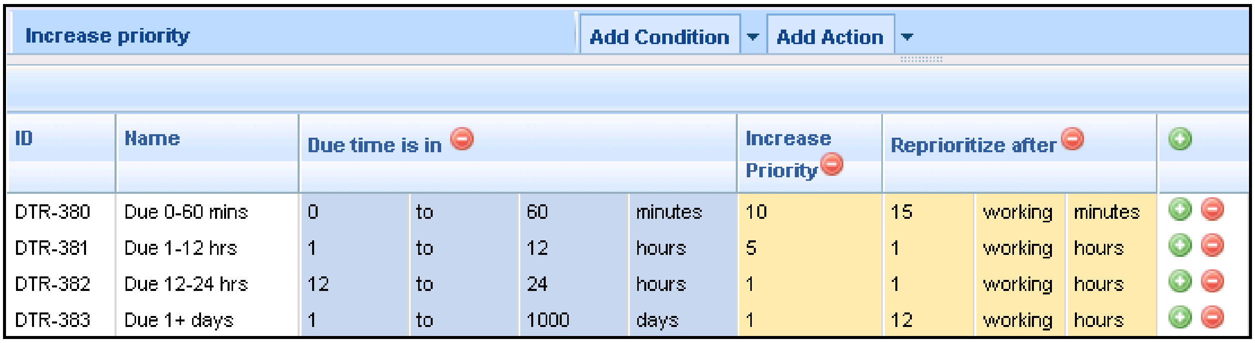
Figure 11: Task Classification Using the "Capture Point is" Rule Condition

For more information about task classification, refer to Appendix C on page 313, specifically "Classification Strategy" on page 320.

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 classification rules and then reapplied after a specified reprioritization period. The reprioritization period is expressed in the same way as any other rule action.



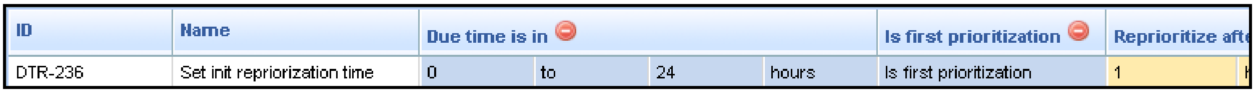
ID	Name	Due time is in				Increase Priority	Reprioritize after		
DTR-380	Due 0-60 mins	0	to	60	minutes	10	15	working	minutes
DTR-381	Due 1-12 hrs	1	to	12	hours	5	1	working	hours
DTR-382	Due 12-24 hrs	12	to	24	hours	1	1	working	hours
DTR-383	Due 1+ days	1	to	1000	days	1	12	working	hours

Figure 12: 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.

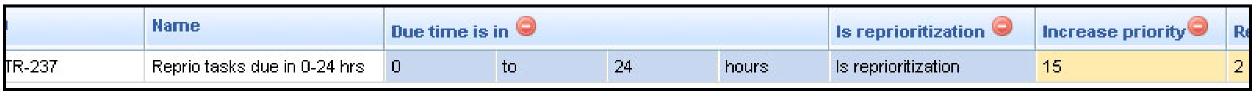
The Standard Rules Template includes two rule conditions, `Is first prioritization` and `Is reprioritization` that should be used in prioritization rules to ensure that the reprioritization interval is set correctly, while avoiding any unnecessary immediate reprioritization of a task (that is, the first 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 15 every 2 hours, if the task is due in less than 24 hours. You want to do the first check 1 hour after the task is classified. You would set this up by using two different prioritization rules, configured in the order shown below. Figure 13 shows the first rule, which includes the `Is first prioritization` condition. The second rule, shown in Figure 14, includes the `Is reprioritization` condition.



ID	Name	Due time is in				Is first prioritization	Reprioritize afte
DTR-236	Set init repriorization time	0	to	24	hours	Is first prioritization	1

Figure 13: Rule 1—Is First Prioritization



	Name	Due time is in ⊖				Is reprioritization ⊖	Increase priority ⊖	R
TR-237	Reprio tasks due in 0-24 hrs	0	to	24	hours	Is reprioritization	15	2

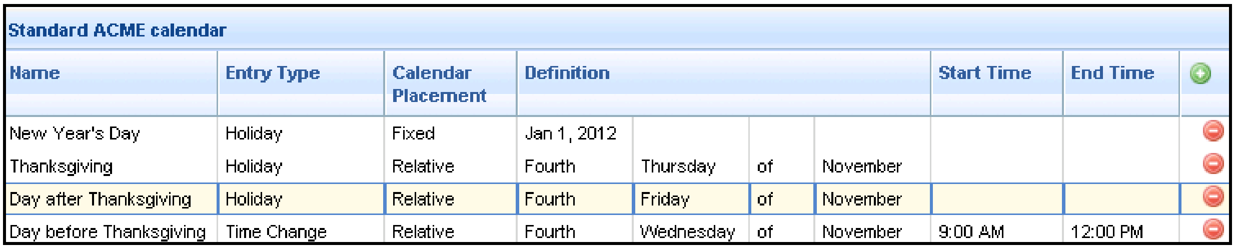
Figure 14: Rule 2—Is Reprioritization

For more information about task prioritization, refer to Appendix C on page 313, specifically “Prioritization Strategy” on page 327.

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 can either be assigned to a rule itself, or can be assigned in a rule action. In either case, the business calendar must be assigned before other rules that use the business calendar can be evaluated. Figure 15 shows a sample business calendar.

Note: In iWD 7.6.1 and 8.0, business calendars were created in iWD Manager. In iWD 8.1, they are created in the Genesys Rules Authoring Tool.



Standard ACME calendar

Name	Entry Type	Calendar Placement	Definition				Start Time	End Time	⊕
New Year's Day	Holiday	Fixed	Jan 1, 2012						⊖
Thanksgiving	Holiday	Relative	Fourth	Thursday	of	November			⊖
Day after Thanksgiving	Holiday	Relative	Fourth	Friday	of	November			⊖
Day before Thanksgiving	Time Change	Relative	Fourth	Wednesday	of	November	9:00 AM	12:00 PM	⊖

Figure 15: Business Calendar

Chapter

4 Installation

This chapter provides information about installing iWD.

This chapter contains the following sections:

Task Summary: Installing iWD 8.1

The following table outlines the task flow for installing iWD 8.1. The procedures in this table provide instructions about installing iWD components on Windows hosts.

Task Summary: Installing iWD 8.1

Objective	Related procedures and actions
1. Prepare for installation and review prerequisites.	• Ensure that your environment meets the prerequisites that are outlined in "Installation Prerequisites" on page 56. • Ensure that the required CDs are available. See "DVD Structure" on page 56.
2. Install Interaction Server.	iWD 8.1 requires Interaction Server 8.1.1 or higher. Genesys recommends that you use the latest version of Interaction Server that is included with iWD. If your environment does not have an instance of Interaction Server installed already (such as for an eServices solution), you must install one. See Procedure: Installing Interaction Server and its databases, on page 62.
3. Install the Genesys Rules System.	Complete steps 1 to 6 (only) in the "Task Summary: Installing Genesys Rules System .1" in the *Genesys Rules System 8.1 Deployment Guide*. **Note:** Step 7 in the Task Summary is about defining the business structure, which is applicable to GRS but not to iWD. You will define the business structure directly in iWD Manager.
4. Install iWD Runtime Node and associate a Person account to the iWD Runtime Node application.	• See Procedure: Installing the iWD Runtime Node (Windows), on page 67. • See Procedure: Associating a Person's account with the iWD Runtime Node application, on page 72.
5. Install iWD Manager	• See Procedure: Creating the iWD Configuration database, on page 73. • See Procedure: Installing iWD Manager (Windows), on page 74.
6. Install iWD Data Mart.	• See Procedure: Installing iWD Data Mart (Windows), on page 86. • See Procedure: Creating the iWD Data Mart Database, on page 92.
7. Configure the necessary Integrated Capture Points for use with iWD.	See the section "Deployment Procedures" in the chapter titled, "Capture Points" in the *eService 8.1 Deployment Guide.*
8. Create an iWD Tenant and iWD Solution in iWD Manager.	See Procedure: Creating an iWD Tenant and iWD Solution in iWD Manager, on page 100.

Task Summary: Installing iWD 8.1 (Continued)

Objective	Related procedures and actions
9. Install and run the iWD Setup Utility to configure various mandatory configuration objects.	• See Procedure: Installing the iWD Setup Utility, on page 104. • See Procedure: Using the iWD Setup Utility, on page 105.
10. Install iWD Stat Server Extensions.	See Procedure: Installing the iWD Stat Extensions, on page 115.
11. Link the iWD components with Genesys Configuration Server.	See Procedure: Creating services in iWD Manager, on page 118.
12. Configure logging for iWD Manager and iWD Runtime Node.	See Procedure: Configuring logging for iWD Manager and iWD Runtime Node, on page 120.
13. Update the Interaction Server databases and Event Log DAP.	See Procedure: Updating the Interaction Server databases and related configuration objects, on page 123.
14. Configure a `Script` object to enable Genesys Rules Authoring Tool to access the iWD configuration database.	See "Enabling Genesys Rules Authoring Tool to Retrieve Dynamic Data from iWD Configuration Database" on page 83
15. Configure remaining iWD services and objects.	See Chapter 5, "Configuration," on page 127.
16. Optional: Configure the system to start/stop application server service through Genesys Solution Control Interface (SCI) or Genesys Administrator.	See Procedure: Using Solution Control Interface or Genesys Administrator to start/stop the application server, on page 59.
17. Optional: Install and configure the sample application to learn more about the functionality of iWD and business rules.	See "Sample Application" on page 126.

Preparing for Installation

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

- "Installation Prerequisites"
- "DVD Structure" on page 56
- "Configuration of Application Servers" on page 57

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

Installation Prerequisites

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

- Refer to "Interaction Server Installation" on page 61 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. For production deployments, install the iWD web applications and the Genesys Rules System web applications on separate instances of the application server.
- A supported database server is installed for the iWD Configuration 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 web site:
  `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 8.1 is compatible with Genesys Universal Routing Server release 7.6.200.04 or higher and Genesys Interaction Routing Designer release 8.1.200.11 or higher.

For information about supported and discontinued platform support, see the *Genesys Supported Operating Environment Reference Manual.*

DVD Structure

All iWD application software is packaged on the iWD DVD. The Genesys Rules System, which is also required for iWD solution installations, is packaged on a separate DVD. The iWD capture points are integrated into Interaction Server, which is packaged on the Interaction Management DVD.

iWD DVD

This DVD 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, includes legacy iWD capture adapter components.
- iWD Manager—A graphical user interface (GUI) for both technical and business configuration of the iWD solution, and includes legacy iWD capture adapter components. iWD Manager is also used for real-time management of tasks.
- iWD Websphere MQ Capture Adapter—An out-of-the-box capture adapter that allows iWD to capture tasks from Websphere source systems.

Note: iWD 8.1 no longer has iWD Capture as a separate installable. It is now included in the iWD Manager and iWD Runtime Node installations.

The iWD Capture software that is included on the iWD DVD is intended for iWD customers who are upgrading to iWD 8.1. For new deployments or existing deployments in which new capture adapters are added, Genesys recommends that you use the Integrated Capture Points, which are integrated with Interaction Server. See also,"Availability of iWD Integrated Capture Points" on page 57.

- 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.

Availability of iWD Integrated Capture Points

The following iWD capture points are available in the following versions of Interaction Server:

- Starting in 8.1.2, the Web Service Integrated Capture Point is available.
- Starting in 8.1.0, the Database Integrated Capture Point is available.
- Starting in 8.0.210, the XML File Integrated Capture Point is available.
- Starting in 8.0.2, the JMS Integrated Capture Point is available.

Configuration of Application Servers

It is necessary to configure your application server to successfully run iWD and the Genesys Rules System. Apache Tomcat and IBM WebSphere are supported.

Note: For production deployments, install the iWD web applications and the Genesys Rules System web applications as separate application server instances, for improved scalability and memory management.

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`
- For iWD Manager, BCMS, and Capture Points, configure the initial and maximum memory pools to `256` and `1024` megabytes. If iWD Data Mart will be running on this particular instance of Tomcat, configure the initial and maximum memory pools to `512` and `1500` (or more, if the operating system will allow it).

If Tomcat is running as Windows console application:

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

  ```
  set JAVA_OPTS=-XX:MaxPermSize=128m -Xms256M -Xmx1536M
  -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 –`Xmx1536M` or more, if allowed by the operating system.

On UNIX machines:

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

  ```
  export JAVA_OPTS="-XX:MaxPermSize=128m -Xms256M -Xmx1536M
  -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 –`Xmx1536M` or more, 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 under the `Tomcat_installation/bin` directory 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.

Starting and Stopping Application Server

To use the Genesys Solution Control Interface (SCI) or Genesys Administrator (GA) to start or stop your web application server, you must complete the following configuration:

Procedure: Using Solution Control Interface or Genesys Administrator to start/stop the application server

Purpose: To configure the Solution Control Interface (SCI) or Genesys Administrator so they can be used to start/stop the web application server.

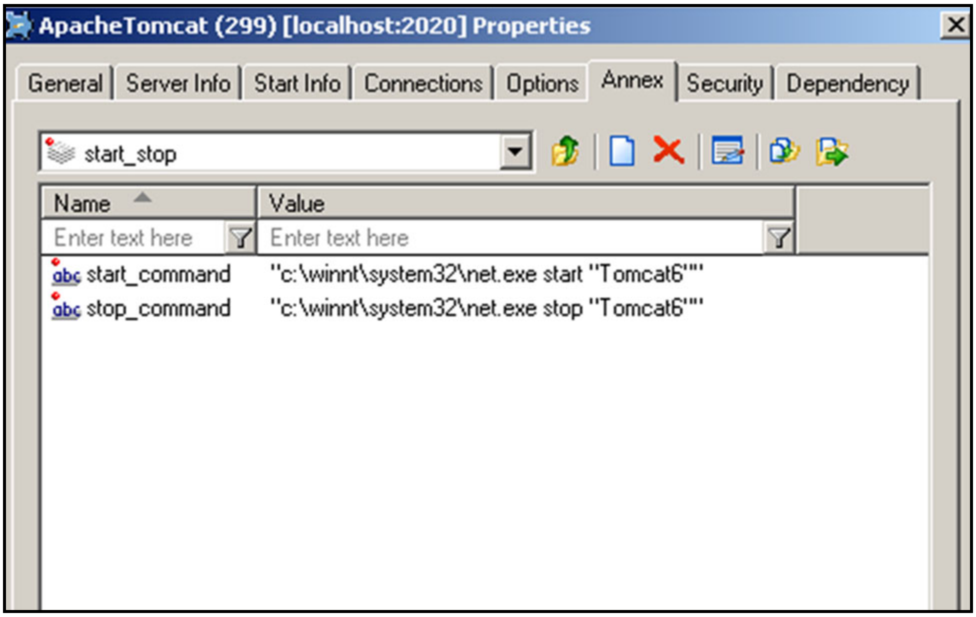
Start of procedure

1. Create a new Configuration Server `Application` of type `3rd Party Server`.
2. On the `Annex` tab of the `Application`, add a new section called `start_stop`.
3. In the new section, add the following two options:
   - `start_command`
   - `stop_command`

4. In the `Value` field, enter the commands to start and stop the Tomcat service, where the name of the Tomcat executable (without the `.exe` extension) is at the end. For example (see also Figure 16):

   ```
   C:\windows\system32\net.exe start "Tomcat6"
   C:\windows\system32\net.exe stop "Tomcat6"
   ```

   Note: Be sure to provide the full path to `net.exe` so that the Genesys Local Control Agent (LCA) can find it.



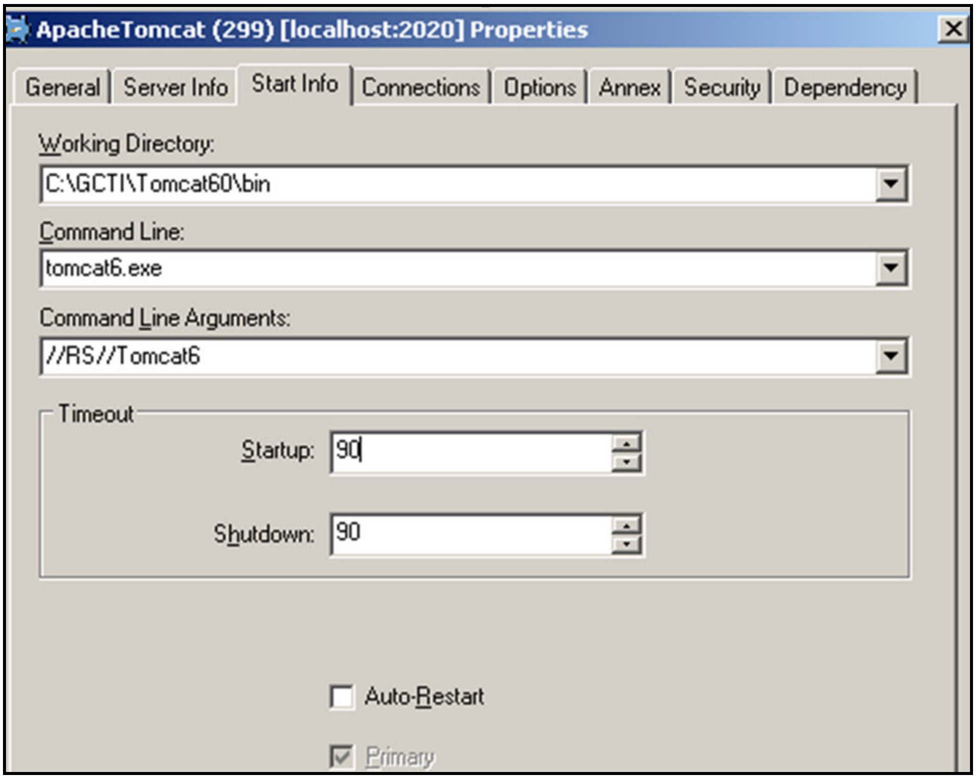
Name	Value
start_command	"c:\winnt\system32\net.exe start "Tomcat6""
stop_command	"c:\winnt\system32\net.exe stop "Tomcat6""

Figure 16: Tomcat Host Properties—Server Info Tab

5. On the `Start Info` tab, set the start command and parameters to match exactly the command and parameter values in the service itself.

   The Solution Control Server (SCS) detects that Tomcat is running by matching the command line parameters of the process that is reported by the LCA with the parameters in Configuration Server. For example (see also Figure 17):

   - `c:\gcti\tomcat60\bin\tomcat6.exe`
   - `//RS//Tomcat6`



ApacheTomcat (299) [localhost:2020] Properties

Working Directory: C:\GCTI\Tomcat60\bin

Command Line: tomcat6.exe

Command Line Arguments: //RS//Tomcat6

Timeout

Startup: 90

Shutdown: 90

□ Auto-Restart

☑ Primary

Figure 17: Tomcat Host Properties—Start Info Tab

Note: In this case, `//RS//` stands for *Run Service* where the service name is `Tomcat6` (the name of the executable). See *Windows service HOW-TO* on the Apache Tomcat 6.0 web site.

End of procedure

Interaction Server Installation

Interaction Server is required for iWD 8.1. 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 (Windows), on page 67. 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 23.

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_eventlog_db`). This database will be used to store event-logging details (refer to the *eServices (Multimedia) 8.1 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 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:evo:oracle:@` rather than `jdbc:oracle:thin:@` or other providers.

   Example: `jdbc:evo: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) 8.1 Reference Manual* for detailed descriptions of the options. Be sure to update the options for the licensing.

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. This option, if it is not already present, will be added for you automatically by using the `Configure Ixn Custom Properties feature` of iWD Manager. (See Procedure: Updating the Interaction Server databases and related configuration objects, on page 123). However, this will only add the `iWD_Completed` queue to the option. You might want to add other queues to this option, based on how you want this `Age` to be calculated. 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 8.1 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 DVD.

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 8.1 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.

Note: To emulate the behavior of iWD 7.6.1 when tasks are assigned to an agent you must configure a specific Interaction Server option. In that release, it was possible to revoke a task from an agent (that is, remove it from the agent desktop even when it was in an `Assigned` state) when Interaction Server received a request from an iWD capture point to `Cancel`, `Complete`, `Hold`, or `Restart` the task. Similarly, it was also possible to `Cancel`, `Hold`, or `Restart` an `Assigned` task, through the Global Task List.

To provide that same behavior in iWD 8.1, add the `enable-revoke-from-agent` option in the `settings` section of the Interaction Server `Options` tab and set the option value to `true`.

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 8.1.2\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

Genesys Rules System Installation

To install Genesys Rules System for use with iWD, complete steps 1 to 6 of the “Task Summary: Installing Genesys Rules System 8.1”, Chapter 2 in the *Genesys Rules System 8.1 Deployment Guide*.

Note: Step 7 of the Task Summary is not required. For iWD installations, you will create the business structure and then, push it to the Genesys Rules System.

For production deployments, install the Genesys Rules System web applications (Genesys Rules Engine and the Genesys Rules Authoring Tool) on a dedicated application server instance, apart from the application server instance that is dedicated to the iWD web applications.

Configuring Genesys Rules System for Use with iWD

To prepare Genesys Rules System for use with iWD, complete the all of the configurations steps described in this section.

Configuring Rule Evaluation Order

To enable Genesys Rules System to evaluate rules for iWD in the expected order, configure two options (one for the Genesys Rules Authoring Tool Server application and one for the Genesys Rules Engine):

1. `sequential-mode`—Set this configuration option value to `false` to ensure the Genesys Rules Engine performs its evaluation in the correct order. Configure this option in the `settings` section in the Genesys Rules Engine `Application`. This is the default setting.
2. `group-by-level`—Set this configuration option value to `true` to ensure the Genesys Rules Engine evaluates all rules within a particular phase, within a particular node of the business structure hierarchy. Configure this option in the `settings` section in the Genesys Rules Authoring Tool server `Application`. This is the default setting.

For more information about the how Rule evaluation order functions, see "Rule Evaluation Order" on page 246.

Creating the Connection to Interaction Server

Create a connection by using the `Connections` tab in the Interaction Server `Application` to the Genesys Rules Engine `Application`.

iWD Runtime Node Installation

This section describes the procedures that are used to install iWD Runtime Node.

Note: The JSF implementation must be set to SunRI1.2 for the `iwd_node` application. To do this:
1. From the installed application list, click on the application.
2. Click on the `[JSP and JSF options]` link.
3. Select `SunRI1.2` from the drop-down list.
4. Click `Save`.

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 166 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 268.

Procedure: Installing the iWD Runtime Node (Windows)

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

Prerequisites

- The environment meets the requirements that are described in “Installation Prerequisites” on page 56.
- The web application server is stopped.
- The iWD DVD.

Start of procedure

1. Log into Configuration Manager or Genesys Administrator and import the iWD Runtime Node `Application` template from the iWD DVD.
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 (`iwdRuntimeNode`) of the iWD DVD.
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:
    a. 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.

    b. In the `Choose Destination Folder` window, browse to the directory in which you want the iWD Runtime Node Java application part to be installed.
    c. After installing iWD Runtime Node and other necessary iWD components, build the WAR archive. See "Post-Installation Steps When Using WebSphere Application Server" on page 93 and install the generated `WAR` file by using the WebSphere Integrated Solutions Console. Usually the WAR archive is generated after the installation of iWD Data Mart.
    d. Skip to Step 13.
12. If you selected Apache Tomcat in Step 10, in the `Choose Tomcat Location` window, 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.
13. Click `Next`, then `Install`, then `Finish`.

End of procedure

Next Steps

- 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 72.

Procedure: Installing the iWD Runtime Node with Tomcat (UNIX)

Purpose: To install an instance of the iWD Runtime Node with Tomcat on the UNIX platform.

Prerequisites

- The environment meets the requirements that are described in "Installation Prerequisites" on page 56.
- The web application server is stopped.
- The iWD DVD.
- You have access rights to execute `install.sh`.

Start of procedure

1. At the root, browse to the install directory and enter `./install.sh`.
2. When the following output is displayed., enter the required information, as indicated at each prompt.

Note: In this procedure Tomcat is select for the servlet container.

```
Installing iWD Runtime Node, version 8.1.xxx.xx
Please select your servlet container type by number:
1. Tomcat
2. WebSphere
=>1
Please enter the full path to your
Tomcat installation =>/iwd81/iwd
Please enter full path of the destination directory for installation
```

```
=>/var/iwd81/runtime
Extracting tarfile: data.tar.gz to directory: /var/iwd81/runtime
...
webapp/
...
webapp/WEB-INF/lib/routingprotocol.jar
webapp/WEB-INF/lib/portlet-api.jar
...
webapp/ui/lib/codepress/codepress.css
webapp/META-INF/
webapp/META-INF/MANIFEST.MF
Installation of iWD Runtime Node, version 8.1.xxx.xx has completed
successfully.
```

End of procedure

Next Steps

- 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 72.

Procedure: Installing the iWD Runtime Node with Websphere (UNIX)

Purpose: To install an instance of the iWD Runtime Node with Websphere on the UNIX platform.

Prerequisites

- The environment meets the requirements that are described in "Installation Prerequisites" on page 56.
- The web application server is stopped.
- The iWD DVD.
- You have access rights to execute `install.sh`.

Start of procedure

1. At the root, browse to the install directory and enter `./install.sh`.
2. When the following output is displayed., enter the required information, as indicated at each prompt.

> **Note:** In this procedure Websphere is selected for the servlet container. When asked to provide the destination directory, enter an arbitrary location. iWD Runtime node Java application and supporting files will be installed in this directory.

```
Installing iWD Runtime Node, version 8.1.xxx.xx

Please select your servlet container type by number:

1. Tomcat

2. WebSphere

=>2

Please enter full path of the destination directory for installation
=>/var/iwd81/runtime

Extracting tarfile: data.tar.gz to directory: /var/iwd81/runtime

iWD_RuntimeNode.sh

support/

support/websphere/

...

webapp/WEB-INF/lib/outboundprotocol.jar

...

webapp/META-INF/

webapp/META-INF/MANIFEST.MF
Installation of iWD Runtime Node, version 8.1.xxx.xx has completed
successfully.
```

End of procedure

Next Steps

- After installation of iWD Runtime node and other iWD components, build the WAR archive as described in Procedure: Building WAR archives for iWD Manager and iWD Runtime Node (UNIX), on page 94 and install the generated WAR file by using WebSphere Integrated Solutions Console.
- 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 72.

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 190 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 or the Server Info section, if you are using Genesys Administrator.
3. In the Log On As section, select This Account and click the Browse button or, if you are using Genesys Administrator, in the Log On Account section, 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 Manager Installation, page 74

Note: The JSF implementation must be set to SunRI1.2 for the `iwd_manager` application. To do this:
1. From the installed application list, click on the application.
2. Click on the `[JSP and JSF options]` link.
3. Select `SunRI1.2` from the drop-down list.
4. Click `Save`.

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.

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

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: Starting with release 8.1, iWD no longer packages the JDBC driver for MySQL. Therefore, if you want to use the MySQL database for the iWD configuration database, you must first obtain and install the JDBC driver.

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 store configuration data and will be accessed by the iWD Runtime Node as well.

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` (Oracle)
   - `CREATE SEQUENCE` (Oracle)

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.

For improved scalability and memory management in production deployments, install iWD Manager and iWD Runtime Node on a separate application server instance from the Genesys Rules System web applications.

Procedure:
Installing iWD Manager (Windows)

Purpose: To install the iWD Manager application on the Windows platform.

Summary

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 73).
- 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 DVD.
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 DVD.
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 (for example, C:\ProgramFiles\Apache\Tomcat60\). The iWD Manager components will be installed in the selected directory, under the webapps/ subdirectory.
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. If you selected Tomcat in Step 7, only supporting files for iWD Manager will be installation in that location. The iWD Manager web application will be installed directly into the webapps folder under your Tomcat home directory.

13. After installation of iWD Manager and other iWD components, build the WAR archive as described in "Post-Installation Steps When Using WebSphere Application Server" on page 93 and install the generated `WAR` file by using WebSphere Integrated Solutions Console.

    You can accept the default or browse to another location on your computer. Click `Next`.

14. 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 73.

15. 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 73).
    - `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`

16. 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`.

17. In the `Ready to Install` screen, click `Install` to begin the installation of iWD Manager.

18. When installation has been completed, click `Finish`.

19. 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 `iwd.properties`

file, which is located in `<web application server directory>/webapps/iwd_manager/WEB-INF/classes` (the password is in plain text in the `iwd.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:\Java\jdk1.6.0_29` then `C:\Java\jdk1.6.0_29\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=iwdmanagerdb
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.SQLServerDialect
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

- Install the remaining iWD components. See "Installing iWD Data Mart" on page 86.
- Run the iWD Setup Utility. See "iWD Setup Utility" on page 102.
- Create a Script object the Genesys Rules Authoring Tool will use to retrieve data from the iWD configuration database. "Enabling Genesys Rules Authoring Tool to Retrieve Dynamic Data from iWD Configuration Database" on page 83.

Procedure: Installing iWD Manager with Tomcat (UNIX)

Purpose: To install the iWD Manager `Application` with Tomcat on the UNIX platform.

Summary

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.

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 73).
- 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.
- You have access rights to execute `install.sh`.

Start of procedure

1. At the root, browse to the install directory and enter `./install.sh`.
2. When the following output is displayed., enter the required information, as indicated at each prompt.

Note: In this procedure Tomcat is selected for the servlet container.

```
Installing iWD Manager, version 8.1.xxx.xx
Please select your servlet container type by number:
1. Tomcat
2. WebSphere
=>1
Please enter the full path to your
Tomcat installation =>/iwd81/iwd

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 =>dbmssql
Please enter the Database name =>iwd810_config
Please enter the Database Server user name =>iwd
Please specify the Database Server user password =>
Please enter the Configuration Server Host Name =>mcr801
Please enter the Configuration Server Port =>2020
Please enter the Configuration Server Backup Host Name =>mcr801
Please enter the Configuration Server Backup Port =>9090
Please enter full path of the destination directory for installation
=>/var/iwd81/manager
Extracting tarfile: data.tar.gz to directory: /var/iwd81/manager
...
webapp/
...
webapp/WEB-INF/
webapp/WEB-INF/application.properties
webapp/WEB-INF/web.xml
webapp/WEB-INF/faces-config.xml
webapp/WEB-INF/lib/
webapp/WEB-INF/lib/hibernate.jar
webapp/WEB-INF/lib/packagedstatisticsdeprecated.jar
webapp/WEB-INF/lib/commons-lang.jar
webapp/WEB-INF/lib/commons-logging.jar
...
webapp/META-INF/MANIFEST.MF
Installation of iWD Manager, version 8.1.xxx.xx has completed
successfully.
```

End of procedure

Next Steps

- Install the remaining iWD components. See "Installing iWD Data Mart" on page 86.
- Run the iWD Setup Utility. See "iWD Setup Utility" on page 102.
- Create a Script object the Genesys Rules Authoring Tool will use to retrieve data from the iWD configuration database. "Enabling Genesys Rules Authoring Tool to Retrieve Dynamic Data from iWD Configuration Database" on page 83

Procedure: Installing iWD Manager with Websphere (UNIX)

Purpose: To install the iWD Manager `Application` with Websphere on the UNIX platform.

Summary

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.

Prerequisites

- The iWD Configuration database has been created (see Procedure: Creating the iWD Configuration database, on page 73).
- 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.
- You have access rights to execute `install.sh`.

Start of procedure

1. At the root, browse to the install directory and enter `./install.sh`.
2. When the following output is displayed., enter the required information, as indicated at each prompt.

Note: In this procedure Websphere is selected for the servlet container. When asked to provide the destination directory, enter an arbitrary location. iWD Manager Java application and supporting files will be installed in this directory.

```
Installing iWD Manager, version 8.1.xxx.xx
Please select your servlet container type by number:
1. Tomcat
2. WebSphere
=>2
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 =>dbmssql
Please enter the Database name =>iwd81_configuration
Please enter the Database Server user name =>iwd
Please specify the Database Server user password =>
Please enter the Configuration Server Host Name =>mcr801
Please enter the Configuration Server Port =>2020
Please enter the Configuration Server Backup Host Name =>mcr801-1
Please enter the Configuration Server Backup Port =>2020
Please enter full path of the destination directory for installation
=>/var/iwd81/manager
Extracting tarfile: data.tar.gz to directory: /var/iwd81/manager
acme/
```

```
acme/.gitignore
...
config/iwdbp.wie
...
webapp/ui/images/services/activeMQService.gif
...
webapp/help/whxdata/whglo.xml
Installation of iWD Manager, version 8.1.xxx.xx has completed
successfully.
```

End of procedure

Next Steps

- After installation of iWD Manager and other iWD components, build the WAR archive as described in "Procedure: Building WAR archives for iWD Manager and iWD Runtime Node (UNIX), on page 94 and install the generated WAR file by using WebSphere Integrated Solutions Console.
- Install the remaining iWD components. See "Installing iWD Data Mart" on page 86.
- Run the iWD Setup Utility. See "iWD Setup Utility" on page 102.
- Create a Script object the Genesys Rules Authoring Tool will use to retrieve data from the iWD configuration database. "Enabling Genesys Rules Authoring Tool to Retrieve Dynamic Data from iWD Configuration Database" on page 83

iWD Configuration Database Auto-Update

iWD Manager can initialize and upgrade its database (referred to as the iWD Configuration 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 Genesys Rules Authoring Tool to Retrieve Dynamic Data from iWD Configuration Database

The values of several types of iWD configuration objects can be retrieved dynamically by the Genesys Rules Authoring Tool (GRAT), during the rules authoring process. Of particular importance, is the fact that GRAT can display a list of configured iWD Capture Points to the rules author, when it is using the `Capture Point is` rule condition. This is one of the out-of-the-box conditions in the iWD Standard Rules Template. It uses a rule parameter of type `database`, which is configured to retrieve the list of Capture Points directly from the iWD configuration database. Part of the configuration of this rule parameter, `capturePoint`, is a profile that instructs GRAT how to connect to the iWD configuration database.

This configuration property called `Profile` corresponds to a `Script` object in Genesys Configuration Server. The script must be created under the `Environment` tenant, because that is where the iWD Standard Rules Template will be migrated to, since it is coming from the `System` tenant in the iWD configuration database.

Give the Script object the name `iWD Manager DB Connection Profile`, in order to match the name used in the iWD Standard Rules Template. The script type must be `Data Collection`. On the `Annex` tab of the `Script` object, add a single configuration section called `database`. In that section, add the following values:

driver

Value: For example, for Microsoft SQL Server 2005 the value might be:
`com.microsoft.sqlserver.jdbc.SQLServerDriver`

Description: The JDBC driver that is used by your database server. For the exact syntax, see the database server documentation.

url

Value: For example, for Microsoft SQL Server, the URL to connect to a database called `IWD_CONFIG` on the host name `mcr810iwd` might be:
`jdbc:sqlserver://mcr810iwd:1433;databaseName=IWD_CONFIG`

Description: The JDBC URL to connect to the iWD configuration database. For the exact syntax, see the database server documentation.

username

Value: Username of database server user with access to the iWD configuration database

password

Value: Password of database server user

After you have added these option, save the script

Note: If you plan to use the Genesys-provided ACME Rules Template in your environment, or access any iWD Lookup Tables from within your business rules, create an identical `Script` object under your child tenant in Configuration Server.

Enabling ADDP Connections Between iWD and the Genesys Suite

To enable the Advanced Disconnect Detection Protocol (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.

Note: You might need to come back to the steps in this procedure after you create these services in iWD Manager. Procedure: Creating services in iWD Manager, on page 118.

End of procedure

Installing iWD Data Mart

When iWD Runtime Node and iWD Manager have been installed, the iWD Data Mart components can be installed.

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

Procedure: Installing iWD Data Mart (Windows)

Purpose: To install an instance of iWD Data Mart for your iWD 8.1 Solution on the Windows platform.

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 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 DVD. 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 8.1 Deployment Guide.*

Procedure: Installing iWD Data Mart with Tomcat (UNIX)

Purpose: To install an instance of iWD Data Mart with Tomcat for your iWD 8.1 Solution on the UNIX platform.

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 iWD Runtime Node are installed.
- You have access rights to execute `install.sh`.

Start of procedure

1. At the root, browse to the install directory and enter `./install.sh`.
2. When the following output is displayed., enter the required information, as indicated at each prompt.

Note: In this procedure Tomcat is the selected servlet container.

```
Installing iWD Manager, version 8.1.xxx.xx
Please select your servlet container type by number:
1. Tomcat
2. WebSphere
=>1
Please enter the full path to your
Tomcat installation =>/iwd81/iwd

Please enter full path of the destination directory for installation
=>/var/iwd81/dm
Extracting tarfile: data.tar.gz to directory: /var/iwd81/dm
config/
config/iwd_reporting.xml
...
etl/aggregate_intraday/1221_select_plugin.ktr
etl/aggregate_intraday/121_set_plugin.ktr
etl/aggregate_intraday/400_etl_audit.ktr
...
lib/sqljdbc.jar
lib/CacheDB.jar
...
sql_scripts/mysql/iwd_dm_mysql.sql
Installation of iWD Data Mart, version 8.1.xxx.xx has completed
successfully.
```

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 8.1 Deployment Guide.*

Procedure: Installing iWD Data Mart with Websphere (UNIX)

Purpose: To install an instance of iWD Data Mart with Websphere for your iWD 8.1 Solution on the UNIX platform.

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.

Summary

When installing iWD Data Mart and Websphere Application Server on UNIX, install iWD Data Mart twice—once to enter iWD Manager installation directory as the target directory, then again, to enter the iWD Runtime node installation directory as the target directory. iWD Manager and iWD Runtime are installed in arbitrary locations and the iWD Data Mart installer cannot locate these directories automatically.

Prerequisites

- iWD Manager and iWD Runtime Node are installed.
- You have access rights to execute `install.sh`.

Start of procedure

Targeting iWD Manager Install Directory

1. At the root, browse to the install directory and enter `./install.sh`.
2. When the following output is displayed., enter the required information, as indicated at each prompt.

Note: In this procedure Websphere is selected as the servlet container.

```
Installing iWD Data Mart, version 8.1.xxx.xx
Please select your servlet container type by number:
1. Tomcat
2. WebSphere
=>2
Please enter full path of the destination directory for installation
=>/var/iwd81/manager
Extracting tarfile: data.tar.gz to directory: /var/iwd81/manager
config/
...
etl/aggregate_stats/
...
sql_scripts/oracle/iwd_dm_oracle.sql
iWD Runtime Node application server directory was not found.
Please install iWD Runtime Node.
```

Targeting iWD Runtime Note Install Directory

3. At the root, browse to the install directory and enter `./install.sh`.
4. When the following output is displayed., enter the required information, as indicated at each prompt.

```
Installing iWD Data Mart, version 8.1.xxx.xx
Please select your servlet container type by number:
1. Tomcat
2. WebSphere
=>2
Please enter full path of the destination directory for installation
=>/var/iwd81/runtime
```

```
Extracting tarfile: data.tar.gz to directory: /var/iwd81/runtime
config/
...
config/iwd_reporting.xml
...
sql_scripts/oracle/
sql_scripts/oracle/iwd_dm_oracle.sql
Installation of iWD Data Mart, version 8.1.xxx.xx has completed
successfully.
```

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 8.1 Deployment Guide.*

iWD Manager on Solaris

If iWD Manager on Solaris fails after logging in, with an error resulting from the locale not being set, use the following commands to set the locale: expected output

Expected Output

```
bash-3.00# LANG=en_US.UTF-8
bash-3.00# export LANG
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=
```

You can also find information on this issue on the following iWD Google site page: `https://sites.google.com/a/iwdlab.com/iwd8/deployment/installation-on-unix-hosts`

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 (Windows), on page 86

Summary

This is a general procedure. Work with your enterprise's database administrator to follow the specific procedure that is required by your database management system and your enterprise policies.

Note: Starting with release 8.1, iWD no longer packages the JDBC driver for MySQL. If you want to use the MySQL database for the iWD Data Mart database, you must first obtain and install the JDBC driver.

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`
   - `CREAT 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 188 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.

End of procedure

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 (Windows)

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 Runtime 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`.

Note: When deploying on Websphere 8.x, the JSF implementation must be set to SunRI1.2 for both the `iwd_manager` and the `iwd_node` applications.

From the installed application list:

a. Click on the application.

b. Click on the `[JSP and JSF options]` link.

c. Select `SunRI1.2` from the drop-down list.

d. Click `Save`.

11. Start the application.

End of procedure

Procedure: Building WAR archives for iWD Manager and iWD Runtime Node (UNIX)

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.sh` 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.sh` file. This will create the `iwd_node.war` file

   The following output will be displayed:

```
bash-3.00# cd /var/iwd81
bash-3.00# ls
manager  runtime
bash-3.00# cd manager/webapps
bash-3.00# ls
iWD_Manager.sh  iwd_manager
bash-3.00# ./iWD_Manager.sh
added manifest
...
adding: ui/css/tooltips.css(in = 613) (out= 613)(stored 0%)
bash-3.00# cd /var/iwd81/runtime/webapps
bash-3.00# ./iWD_RuntimeNode.sh
added manifest
...
adding: ui/lib/codepress/images/line-numbers.png(in = 16556) (out=
16556)(stored 0%)
bash-3.00#
```

5. Complete Steps 5 to d in the Procedure: Building WAR archives for iWD Manager and iWD Runtime Node (Windows), on page 93.

End of procedure

Integrated Capture Points

Capture point functionality allows iWD to create new tasks, based on data coming from an enterprise application or *source system*. Capture points also enable existing tasks to be canceled, completed, held/resumed, restarted, or modified.

Capture point functionality is integrated into Interaction Server. However, for customers upgrading from iWD 7.6.1 or 8.0, iWD 8.1 also supports the *legacy* capture point services from those releases.

Genesys recommends that you use the Integrated Capture Points. Each of these supports some level of backward compatibility for those customers that are accustomed to using the legacy capture point services. For more information about how the Integrated Capture Points compare to the legacy capture point services, see “Moving from Legacy iWD Capture Points to Integrated Capture Points”.

Installing iWD Integrated Capture Points

Installation procedures for Integrated Capture Points are described in Chapter 12 of the *eServices 8.1 Deployment Guide.*

The prerequisites and other information will differ in the eServic*es 8.1 Deployment Guide*, depending on the Integrated Capture Point being used. However, for all Integrated Capture Points there are two common installation steps:

1. Creating the capture point application. See the section “Creating the capture point application” in the *eServices 8.1 Deployment Guide.*
2. Configuring the Generic Capture Point service. See Procedure: Creating services in iWD Manager, on page 118.

Moving from Legacy iWD Capture Points to Integrated Capture Points

This section is intended to help you understand how the native Interaction Server XML message operations compare to the native iWD XML message operations. The information provided here is mainly derived from the *eServices 8.1 User's Guide.*

Mapping Native iWD Task Actions to Native Interaction Server Message Operations

Table 2 shows how the operations you would specify in the iWD message format correspond, or translate, to the operations you would specify in the Interaction Server message format.

Integrated capture points can also work in compatibility mode with legacy capture points. Please see the Interaction Server documentation set for more information.

Table 2: Mapping Native iWD Task Actions to Native Interaction Server Message Operations

Task action	iWD XML message operation	Interaction Server message operation
Create a new task/interaction in iWD	`CreateTask`	`<interaction operation="submit">`
Get information about a task/interaction in iWD	`GetTaskInfo`	`<interaction operation="getinfo">`
Update a task/interaction in iWD	`UpdateTask`	`<interaction operation="update">`
Complete a task/interaction in iWD	`CompleteTask`	`<interaction operation="update">`
Hold a task/interaction in iWD	`HoldTask`	`<interaction operation="hold">`
Resume a held task/interaction in iWD	`ResumeTask`	`<interaction operation="resume">`
Restart a task/interaction in iWD	`RestartTask`	`<interaction operation="update">`
Cancel a task/interaction in iWD	`CancelTask`	`<interaction operation="update">`

Creating a New Task or Interaction

When you are creating a new task/interaction by using the `<interaction operation="submit">` operation to specify the `all initial` properties of the interaction, use the `properties` element, which is a direct child of the `interaction` element. Alternatively, you can provide default values for any interaction properties, as part of the Integrated Capture Point `Application` configuration. See the descriptions of the default-values sections (which are repeated for each capture type) in the *eServices 8.1 Reference Manual.*

Configuration of some properties is mandatory to ensure iWD behaves as expected. See “Mandatory Interaction Properties” on page 97.

Also, in Table 43 on page 272, in the column "Interaction Attached Data Key" the names of the properties are described as you would refer to them in this message. You can cross-reference this with the information in the tables titled, "Translation Table for Known Attributes-Inbound" and "Translation Table for Known Attributes-Outbound" in the *eServices 8.1 User's Guide.*

Updating, Completing, Canceling, or Restarting a Task/Interaction

Use the `<interaction operation="update">` operation when you are updating, completing, canceling, or restarting a task/interaction. Use the `properties` element, which is a direct child of the `interaction` element, to specify which properties to update.

In the special cases, when you are completing, canceling, or restarting a task/interaction, specify the `Queue` property, to move the task/interaction to the `iWD_Completed` queue, `iWD_Canceled` queue, or the `iWD_New` queue, respectively.

Note: When you use the `iWD_Completed`, `iWD_Canceled`, and `iWD_New` queue names, it is assumed that you are using the out-of-the-box business process that is provided with iWD (IWDBP). If you have modified this business process to add/change queue names, you will need to consider this when you are specifying the `Queue` property for the update action.

Mandatory Interaction Properties

When you submit a new interaction to Interaction Server (`<interaction operation="submit">`), the following properties are mandatory. Therefore, they are also mandatory for the Integrated Capture Points:

- `InteractionType`
- `InteractionSubtype`
- `MediaType`
- `Queue`
- `TenantID`

Note: In this case, `TenantID` is the Genesys tenant ID, not the ID of the managed tenant that is created in the iWD configuration.

You do not need to specify these properties in the XML message, because several of them are normally not relevant to iWD. There is an configuration option for the Capture Point `Application` template, called `default-values` (for a description, see the *eServices 8.1 Reference Guide*), which enables you to enter default values for any of these properties (as well as any other interaction properties). Therefore, if the default values for these five attributes/properties are specified in the options, then the message itself need not contain any other properties for Interaction Server to accept and process it.

The Queue property is an exception, because you can configure the default queue as part of the business process configuration by using the endpoints section.

In addition to these mandatory properties, three other properties, specific to iWD interactions, are used to ensure that interactions are accounted for correctly by iWD Data Mart and are displayed properly through the Global Task List. The properties are:

- iWD_TenantId
- iWD_SolutionId
- iWD_CapturePointId

The iWD_CapturePointId property will be equivalent to the name of your Capture Point Application in Configuration Server. See the procedure, “Creating the capture point application” in the *eServices 8.1 Deployment Guide*.

When you deploy the Integrated Capture Point, you must also create a Generic Capture Point service in iWD Manager, to represent the Integrated Capture Point. To create a Generic Capture Point, see Step 7 in the Procedure: Creating services in iWD Manager, on page 118.

The iWD_SolutionId and the iWD_TenantId properties are the ID of the Solution and the Tenant, respectively, under which the capture point service has been created in iWD configuration. Similar to the mandatory interaction properties, these three additional properties can be set in the default-values section of the Capture Point Application template, or they can be specified as part of the submit action.

Other Information About Capture Points To Consider

This section provides some additional information about Capture Points that you might want to consider.

For a description of some optional ways to format the information inside the <properties> element, see the section, “Properties Element” in the *eServices 8.1 User's Guide.*

For additional information about other elements, such as changed, deleted, reason, actor, and party, and about how responses to capture point requests are processed, see Chapter 9 in the *eServices 8.1 User's Guide.*

By default, when you specify any interaction property that is not considered a core property of the Interaction Server interaction data model, or that is not a core or extended attribute in terms of the iWD data model, the value of that property will be stored in a BLOB in the interactions table of the Interaction Server database. If you need to use this custom property in any Queue Views, or to use filtering in the Global Task List, follow the Procedure: Configuring a custom interaction property, on page 276.

Importing iWD Configuration XML Files

Procedure: Import default configuration XML files into iWD Manager

Prerequisites

- iWD Manager, iWD Runtime Node, and iWD Data Mart are installed.
- Application servers are started (if running on WebSphere – applications are started).

Start of procedure

1. Open this link in a web browser:
   `http://app_server_host:port/iwd_manager/`
2. Perform Create/Upgrade Configuration Database from login screen
3. Log in to iWD Manager using a username 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`.
4. Enter the `CME Application Name`.

   Enter the name that you gave your iWD Manager `Application` object in Step 2 of Procedure: Installing iWD Manager (Windows), on page 74. For notes about how to enable a user to log in to iWD Manager, see "General Conditions for Configuring an iWD Manager User" on page 129.
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`.
   - `<iWD Manager installation directory>\config\iwd_capture.xml`
   - `<iWD Manager installation directory>\config\iwd_transform.xml`

   Note: The XML files are saved to the installation directory where the iWD Manager and iWD Data Mart supporting files were installed.

   - Data Mart:
     `<iWD Datamart installation directory>\config\iwd_reporting.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>:<serverport>/iwd_manager/`
      where `<servername>` server is the name of the server on which iWD Manager is installed and `<serverport>` is the application server port.
   b. Log into iWD Manager.
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 127 for more information about the iWD Manager user interface).
3. In the navigation tree, select `New` under `Managed Tenants`.

Create the iWD Tenant

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 155 for more information about iWD Tenants. Remember that tenants in iWD correlate one-to-one with tenants in Configuration Server.

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.

Create the iWD Solution

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 167 for information about the various properties than can be configured for an iWD Solution). Click Save.

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 168 for information about the various properties that can be configured for a runtime node). In production environments users should create a dedicated runtime node for the Data Mart services as well.

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 you create, run the iWD Setup Utility and install the Stat Server Java Extension.

> **Note:** Each iWD solution requires its own iWD Data Mart.

iWD Setup Utility

Installing and running the iWD Setup Utility is the next step in installing iWD 8.1. The iWD Setup utility performs the following functions:

- Imports iWD business processes to Genesys Configuration Server. The iWD Setup Utility includes the following business processes:
  - IWDBP—the default business process, explained in detail in Appendix C, "iWD Business Process (IWDBP)," on page 313.

  - Standard Genesys to IWD adapter—the business process used to insert into IWDBP to serve Genesys standard open media channels, explained in Appendix D, "Adapting the iWD Business Process for Standard Genesys Channels," on page 343.

    Note: These business processes are also provided as a `.wie` file that can be imported manually through the Genesys Interaction Routing Designer. This is useful when you are upgrading from one iWD release to another and you do not have to run the iWD Setup Utility. The iWD business process `.wie` file is saved to your file system when the iWD Manager Installation Package is installed.

  - ABC IWD Simple BP—the business process used to insert into existing business processes, explained in Appendix E, "iWD Rules and Existing Business Processes," on page 367.
- Creates 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 the capacity rule that includes the media type `workitem` and provides the option to assign it to the tenant.
- Enables you to select and configure the Stat Server to use with iWD Stat Extensions.
- Configures the iWD Stat Server Java Extensions into the specified Stat Server.
- Creates the two List Objects—`Iwd_Esp_List` and `Iwd_Package_List`—that are necessary to ensure business rules are invoked from business processes, such as IWDBP.

  Note: In release 8.1.1, `Iwd_Esp_List` also defines the list of Universal Contact Servers to be used with Interaction Server that allow the iWD business process (IWDBP) to update the interaction record in the Universal Contact Server, to mark the interaction as *done*.

- 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 DVD 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 Server 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.

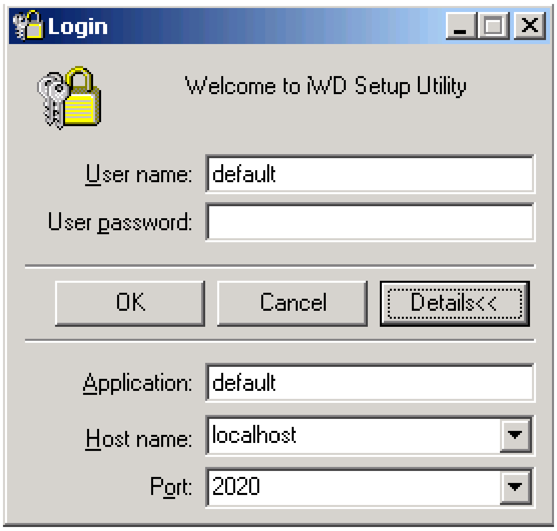
If you have made changes to the iWD business process, but would like to see the business process that is included in the iWD Setup Utility, you must export the customized business process from IRD, delete it, and then run the iWD Setup Utility again. Or, you can run the iWD Setup Utility against a different 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 (Windows), on page 74 and Procedure: Installing the iWD Runtime Node (Windows), on page 67.
- The iWD Data Mart has been installed and the iWD Data Mart database has been created. See Procedure: Installing iWD Data Mart (Windows), on page 86 and Procedure: Creating the iWD Data Mart Database, on page 92.
- 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 Installation” on page 61.

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 18), and click `OK`.



Login

Welcome to iWD Setup Utility

User name: default

User password:

Application: default

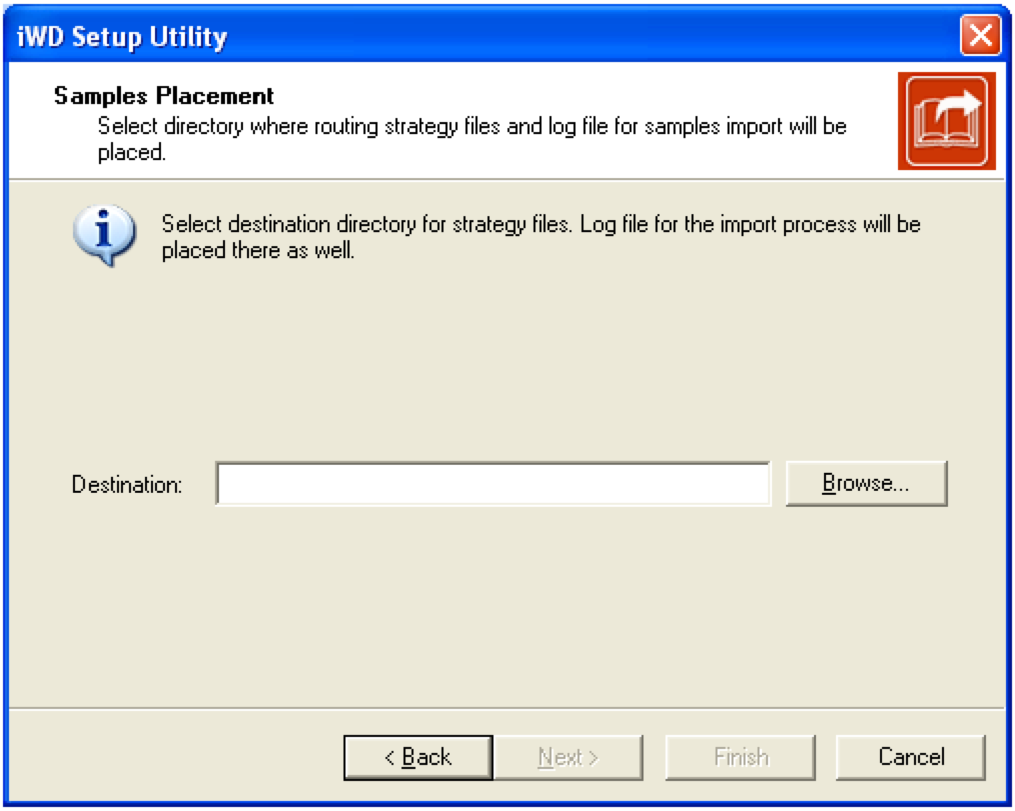
Host name: localhost

Port: 2020

Figure 18: 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 19) is displayed. Browse to the desired destination directory for strategy files and click `Next`.



iWD Setup Utility

Samples Placement

Select directory where routing strategy files and log file for samples import will be placed.

Select destination directory for strategy files. Log file for the import process will be placed there as well.

Destination:

Figure 19: iWD Setup Utility Samples Placement screen

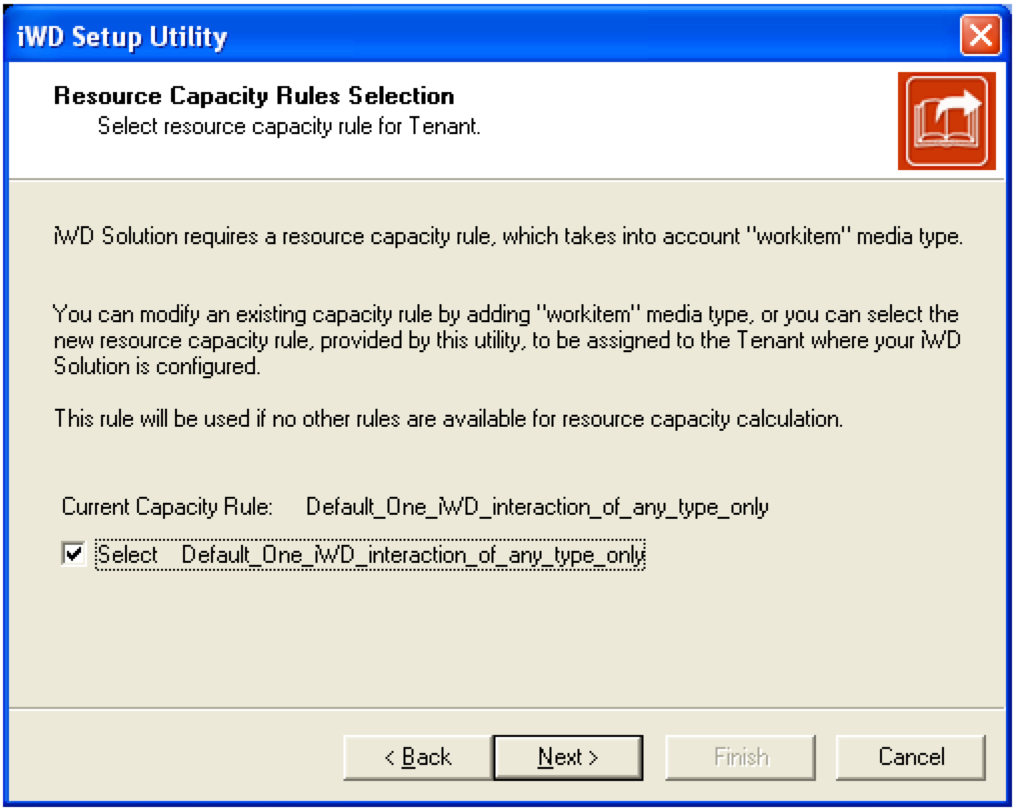
6. The iWD Setup Utility imports the configuration objects and relocates the strategy files. On the `Import Completed` screen, click `Next`.

7. The `Resource Capacity Rules Selection` screen is displayed (see Figure 20). 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 Points that you create 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 Points you create, as well as in any capacity rules you configure.

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

- *Universal Routing 8.1 Business Process User's Guide*
- *Universal Routing 8.1 Reference Manual*
- *eServices (Multimedia) 8.1 User's Guide*



iWD Setup Utility

Resource Capacity Rules Selection

Select resource capacity rule for Tenant.

iWD Solution requires a resource capacity rule, which takes into account "workitem" media type.

You can modify an existing capacity rule by adding "workitem" media type, or you can select the new resource capacity rule, provided by this utility, to be assigned to the Tenant where your iWD Solution is configured.

This rule will be used if no other rules are available for resource capacity calculation.

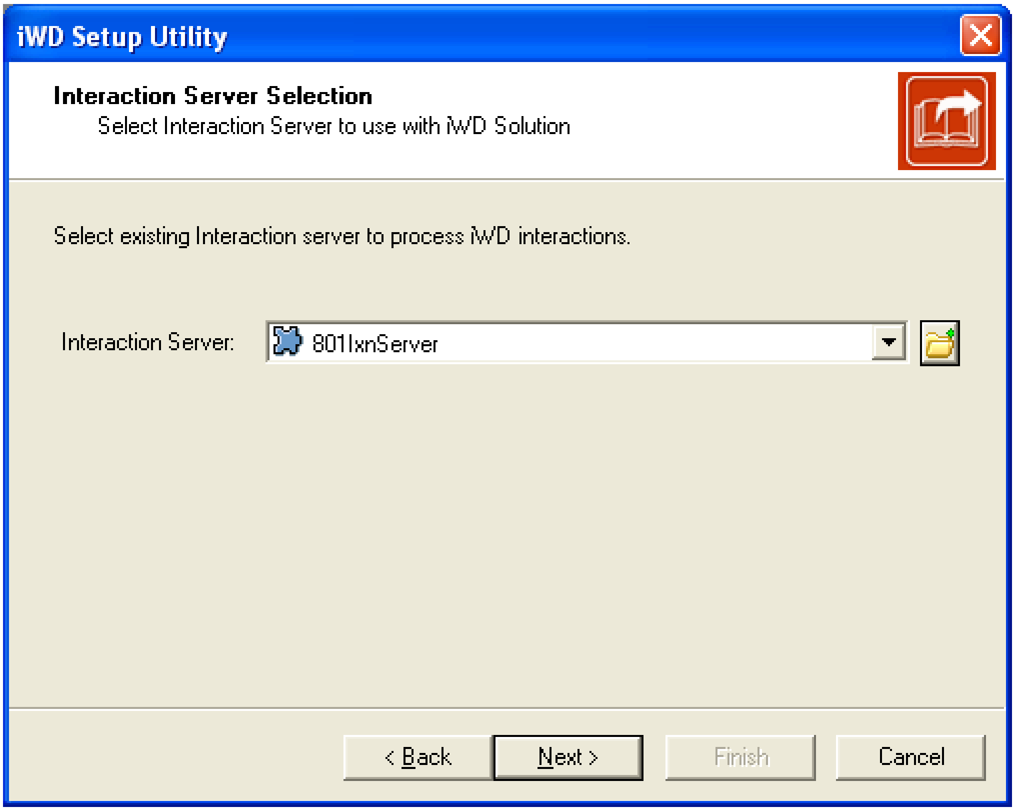
Current Capacity Rule: Default_One_iWD_interaction_of_any_type_only

☑ Select Default_One_iWD_interaction_of_any_type_only

Figure 20: iWD Setup Utility Resource Rules Selection screen

Note: The iWD Setup Utility Resource Rules Selection screen (shown in Figure 20) 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 107.

8. On the `Interaction Server Selection` screen (see Figure 21), browse to the Interaction Server you plan to use for this iWD Solution. Click `Next`.



iWD Setup Utility

Interaction Server Selection

Select Interaction Server to use with iWD Solution

Select existing Interaction server to process iWD interactions.

Interaction Server: 801IxnServer

Figure 21: 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 62.

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 8.1 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.

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

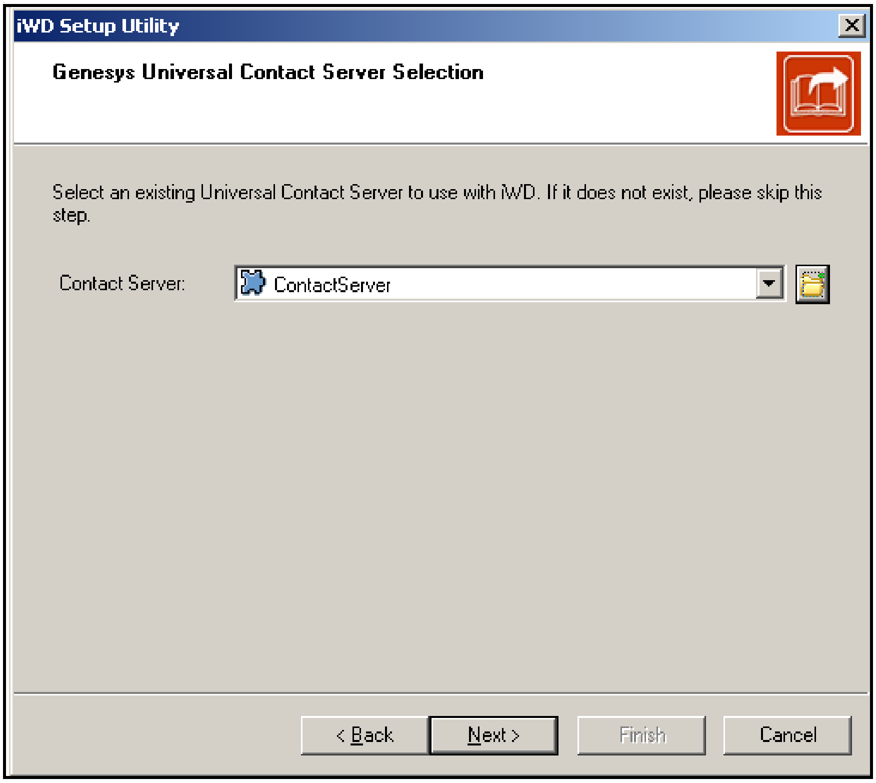
    `Database Name`: Enter the name of the iWD Data Mart database (or the SID if using an Oracle database).

    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) 8.1 Deployment Guide* and *eServices (Multimedia) 8.1 User's Guide* for more information about Multimedia Switches. Click `Next` to display the Universal Contact Server Selection screen.



iWD Setup Utility

Genesys Universal Contact Server Selection

Select an existing Universal Contact Server to use with iWD. If it does not exist, please skip this step.

Contact Server: ContactServer

Figure 22: Universal Contact Server Selection Screen

13. On the `Genesys Universal Contact Server Selection screen` (Figure 22) select an existing Genesys Universal Contact Server (UCS) that is connected to Interaction Server. Having the Universal Contact Server name associated with your iWD Solution ensures that the business logic in the out-of-box iWD business process (IWDBP) will update the interaction record in the UCS database when an interaction is considered *done* (that is, the value of the `Status` column in the `Interaction` table in the UCS database will be set to `3`).

    Normally such an update is done in a routing strategy when processing is stopped for an interaction and the interaction is deleted in the Interaction Server database. However, in the case of the iWD business process, even though interactions may persist in the Interaction Server database for a long period of time, they should still be considered *done* from an interaction history standpoint in the UCS database.

    Selecting a Universal Contact Server application on this screen will cause the iWD Setup Utility to update the `Iwd_Esp_List` list object with an association between your iWD Solution name and this UCS application name.

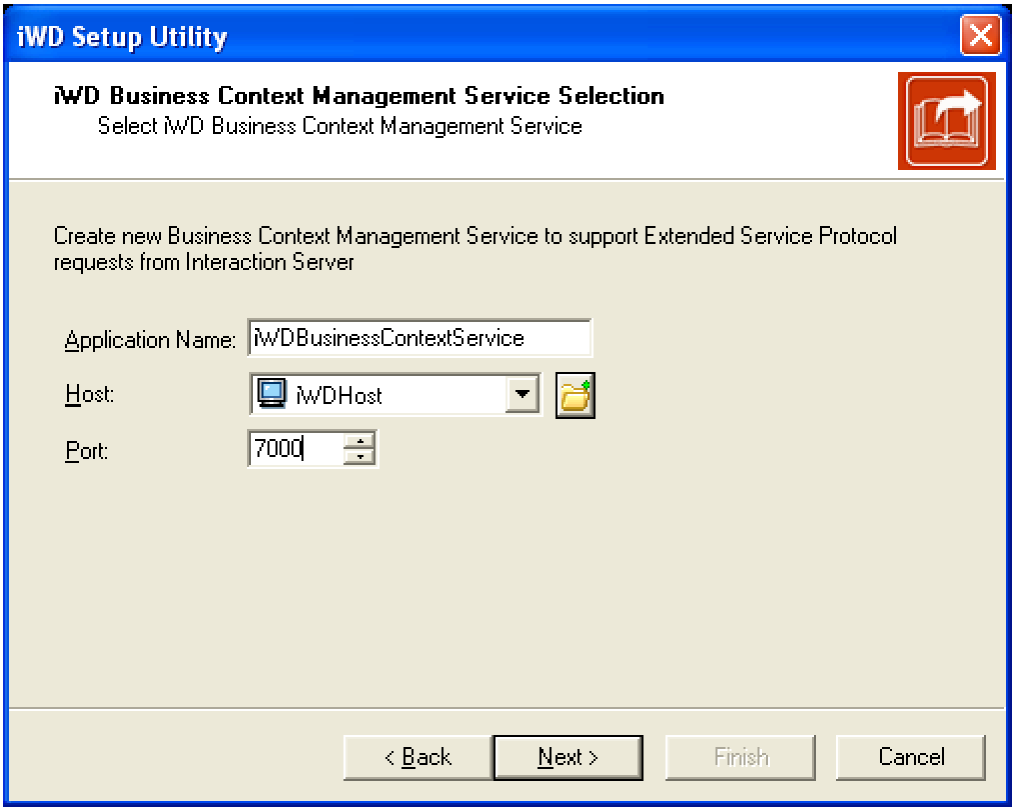
14. On the `iWD Business Context Management Service Selection` screen (see Figure 23), enter the following information to create a new iWD Business Context Management Service to support Extended Service Protocol (ESP) requests from Interaction Server and then, click `Next:`

- `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.

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

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

Refer to page 266 for information about configuring multiple Business Context Management Services.



iWD Setup Utility

iWD Business Context Management Service Selection

Select iWD Business Context Management Service

Create new Business Context Management Service to support Extended Service Protocol requests from Interaction Server

Application Name: iWDBusinessContextService

Host: iWDHost

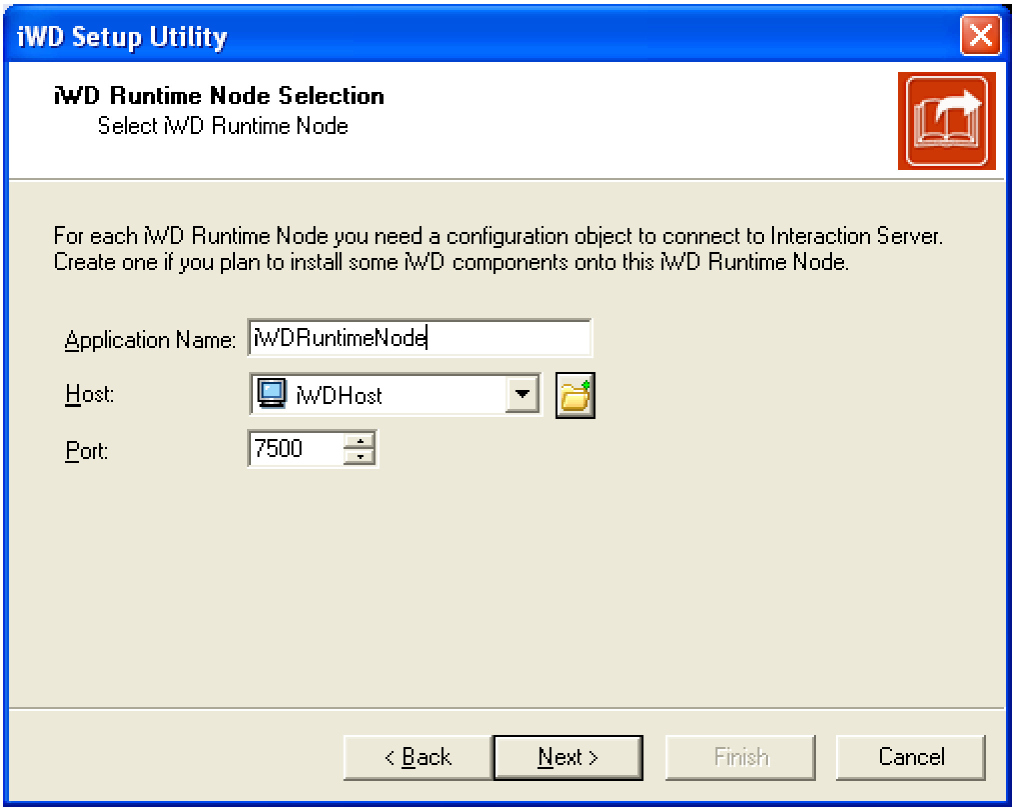
Port: 7000

Figure 23: iWD Business Context Management Service

15. On the next screen (see Figure 24), enter the following information to create an iWD Runtime Node application and then, click `Next`.
    - `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.

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



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.

Application Name: iWDRuntimeNode

Host: iWDHost

Port: 7500

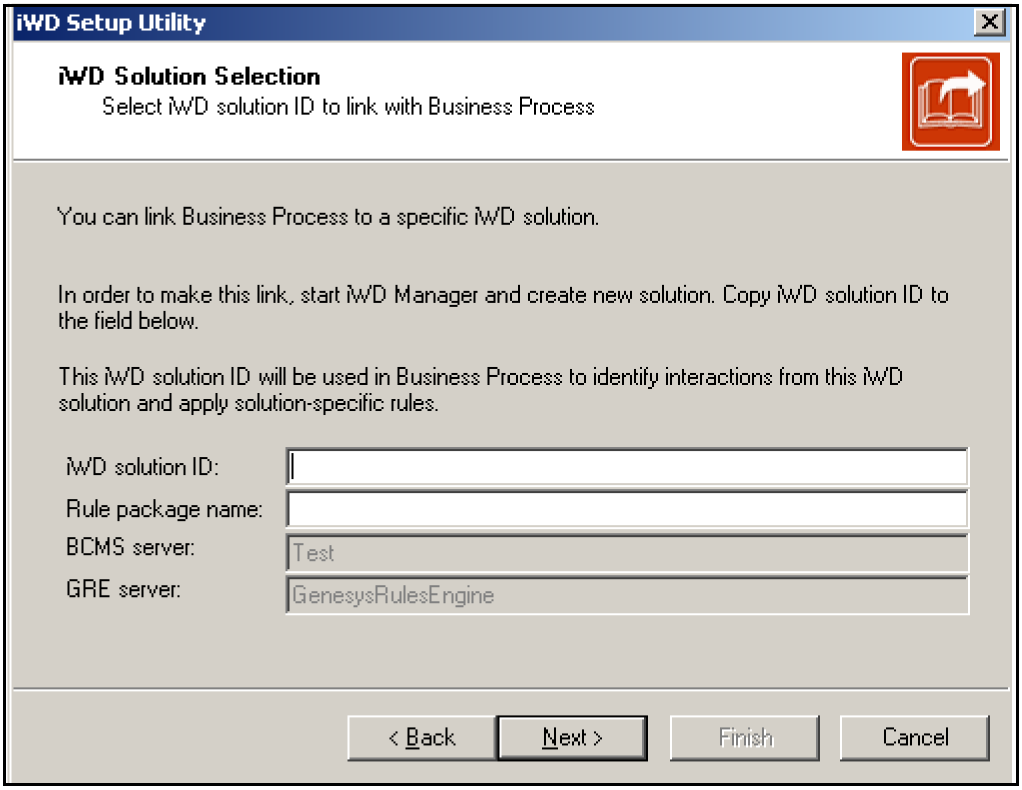
Figure 24: iWD Runtime Node

16. On the `Genesys Rules Engine Application Selection` screen, select the rules engine that will evaluate the iWD rules and click `Next`.
17. On the `iWD Solution Selection` screen, enter the Solution ID for the solution that you created in Procedure: Creating an iWD Tenant and iWD Solution in iWD Manager, on page 100.

18. In the Rule package name field, enter the name of the rule package that will be evaluated by the Genesys Rules Engine, that maps to the IWD Solution.

Note: If you do not know the name of the rule package yet, you still need to enter some value in this field, which you can update later. These values will be used by the Setup Utility to create two List Objects—Iwd_Esp_List and Iwd_Package_List. These List Objects are necessary ensure business rules are invoked from business processes, such as IWDBP. If you leave the Rule package name field empty, the Setup Utility will not create the two List Objects successfully.

The ESP server field is disabled, but it displays the name of the Business Context Management Service that you created in Step 14 (see Figure 25). Click Next.



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.

iWD solution ID:

Rule package name:

BCMS server: Test

GRE server: GenesysRulesEngine

Figure 25: iWD Solution Selection

19. Click Finish to complete the iWD Setup Utility.

End of procedure

Next Steps

- Install iWD Stat Extensions.

Enabling/Disabling ADDP for BCMS

Procedure:
Enabling/Disabling ADDP for Business Context Management Services (BCMS)

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

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 provide access to the aggregated data in the Data Mart.

Note: If the Data Mart is not used, Stat Extensions are not required and the solution will work fine.

Prerequisites

- An instance of Stat Server is installed, dedicated for use with iWD. Refer to the *Framework 8.1 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 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 the 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.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: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.
- `service-id-1`: The runtime ID of the Statistics Adapter service in the iWD configuration.
- `service-tenant-1`: The name of the CME tenant, that is linked to the iWD managed tenant where the Statistics Adapter service is configured.
- `tenant-ids`: A list of Configuration Server/Genesys Administrator tenant names mapped to iWD tenant IDs, separated by a comma. For a system with one tenant with name `TenantA` and ID `T2`, this option should be set to value:
  `Environment=1,TenantA=2`

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, Generic Capture Point, 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 178. 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 (Windows), on page 74.
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 183 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 184 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 185 for information about the properties and possible values.

   f. Click Save to save the new service.

7. Create a Generic Capture Point Service (if integrated capture points are used).

   a. Select Services > New Service under the solution.

   b. Enter a name for the service.

   c. From the Service Template drop-down list, select Generic Capture Point.

   d. Modify the Service ID property to match the name of the Integrated Capture Point application you created in Configuration Manager or Genesys Administrator (see “Integrated Capture Points” on page 95.)

   e. Enter a description for the service.

   f. Configure values for the rest of the properties. For information about the properties and possible values, see “Generic Capture Point Service” on page 188.

   g. To save the new service, click Save.

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 187 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 266 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 though 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 and Related Configuration Options" on page 123.

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 8.1 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 file 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 182 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.AbstractFlushingEventList
ener=FATAL
log4j.category.org.apache.commons.httpclient=ERROR
log4j.category.org.apache.commons.digester=ERROR
log4j.category.org.codehaus.xfire.transport.http.HttpChannel=FAT
AL
log4j.category.org.codehaus.xfire.handler.DefaultFaultHandler=ER
ROR

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 and Related Configuration Options

After the iWD components have been installed, some updates are required for the Interaction Server database, Interaction Server Event Log database, and Interaction Server Event Log Database Access Point options. In addition, new Business Attributes must be created in Configuration Server. Most of these tasks are performed automatically through iWD Manager.

Procedure: Updating the Interaction Server databases and related configuration objects

Purpose: To ensure iWD automatically runs the update scripts on the Interaction Server databases for compatibility with iWD 8.1, updates the configuration options for the Event Log DAP, creates the necessary Business Attributes in Configuration Server, and configures the `completed-queues` option for Interaction Server.

Prerequisites

- Interaction Server is installed as described in Procedure: Installing Interaction Server and its databases, on page 62 (also refer to the *eServices (Multimedia) 8.1 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.

Start of procedure

1. Log into iWD Manager.
2. Navigate to your iWD managed tenant.
3. From the navigation tree, select `Services` and expand your iWD solution.
4. Select `Configure Ixn Custom Properties.`

   The `Mapping errors` section on the right-side pane notifies you of the updates that must be made to your Interaction Server Event Log DAP and Business Attribute configuration. The `Interaction Server & Event log database migrations` section notifies you of the updates that must be done to your Interaction Server and Interaction Server Event Log databases.
5. Initiate these updates; click `Configure Ixn Custom Properties.`

   You are prompted to restart Interaction Server.

End of procedure

enable-revoke-from-agent

Starting with Interaction Server 8.1.2, a configuration option, `enable-revoke-from-agent,` has been provided to support enhancements to task management in the Global Task List. Specifically, this option allows an `Assigned` interaction to be revoked from an employee desktop when the interaction is put on hold, canceled, or completed, from the Global Task List or through an iWD capture point. See Chapter 7 on page 249.

Preparing iWD For Use With Genesys Social Engagement

By default, the installation and configuration procedures described above will prepare iWD to support Genesys Social Engagement. That is, the required database fields will be present in the Interaction Server and Interaction Server Event Log databases, the required Interaction Custom Properties (Business Attributes) will be created in Configuration Server, and the required Interaction Server and Event Log Database Access Point application options will be configured.

However, a specific configuration option must be configured in order for the social media-related attributes (such as, `Actionable`, `Influence`, `Message`, `Name`, `Post Date`, and `Sentiment`) to be visible in iWD's Global Task List. Configuring this option as described in the Procedure: Updating the Tenant to enable display of Social Media attributes and filters will have the following effect:

- A Global Task List custom filter that uses specific social media-related columns will be visible in the `Filters` drop-down list.
- Social media-related attributes will be available custom filters in the advanced `Filters` drop-down list of the Global Task List.
- Social media-related attributes will be visible on the `Attributes` panel of the Global Task List.

Procedure: Updating the Tenant to enable display of Social Media attributes and filters

Purpose: To update the tenant to enable iWD to display social media attributes and filters.

Start of procedure

1. Log in to Configuration Manager or Genesys Administrator.
2. Open the properties of the Configuration Server tenant that maps to the iWD managed tenant with which you want to use the Genesys Social Engagement solution.
3. In the `Annex` tab of the tenant, create a new section named `iwd.profile`.
4. In the `iwd.profile` section, create a new option named `social-messaging`.
5. In the value field, enter `true`.
6. To save the configuration, click `Save`.

End of procedure

Next Steps

Now you should start configuring the mandatory and optional services for your Solution. Go to “iWD Services” on page 178 for information about the available services, their recommended order of configuration, and the properties that can be configured for each service. If you want to review how to work with iWD Manager first, start at the beginning of Chapter 5.

Sample Application

iWD 8.1 includes a sample application (ACME) that can be used to learn about and experiment with iWD. The sample application is included on the iWD 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: If you want to set up the ACME web site to create tasks, the `ACME` sample application requires MySQL. If you do not want to set up the ACME web site, you can create tasks through XML.

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

5 Configuration

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

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 72 of Chapter 4). 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.

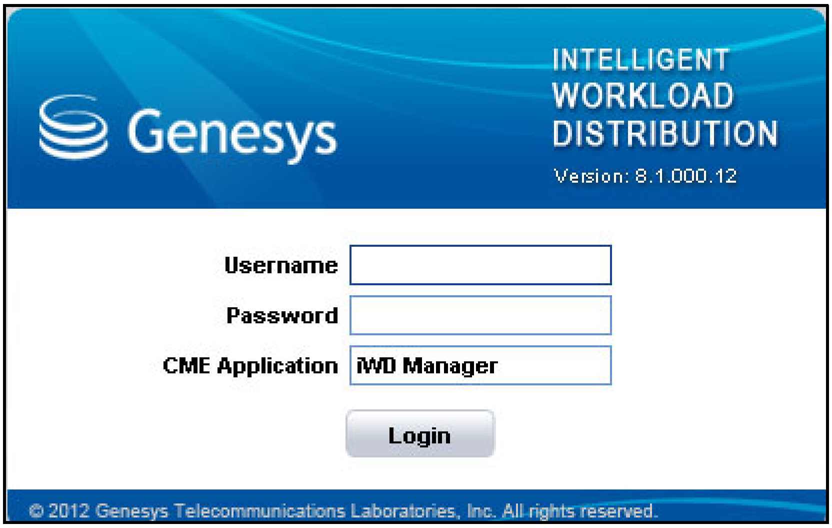
iWD Manager Login

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).

The login screen (see Figure 26) 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.



CME Application: iWD Manager

Figure 26: 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=<CMEApplicatio
n>&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=jsmi
th&password=myPassword&CMEApplication=iWDManager
```

General Conditions for Configuring an iWD Manager User

- For a user to have access to Tenants, you must configure, in the Security Policy section of iWD Manager, a role mapped to the iWD Access Group.
- 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` and `Execute` 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 valueiWDManagerPlace 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

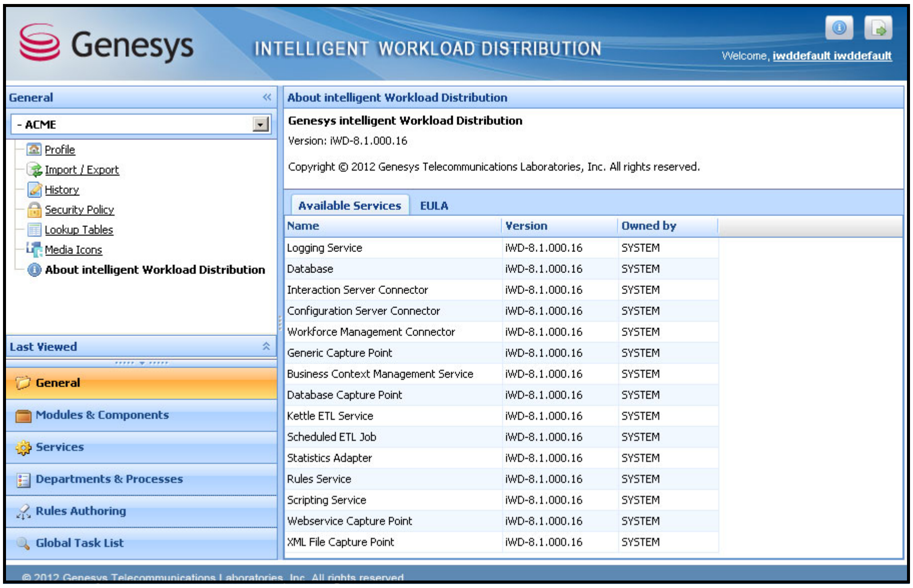
- You want the default user to also have full access to iWD Manager, including the ability to not only view tasks in the Global Task List under the IWD_ACME managed tenant, but also to manage the tasks through the Global Task List (Cancel, Hold/Resume, Modify)
- You change his Employee ID to an Employee 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 User Interface

The iWD Manager user interface, shown in Figure 27 on page 133, is composed of three main application areas:

- Header bar (see page 133). Located on top, it shows general information and actions, such as logged-in user name and logout action.
- Navigational panel (see page 134). Located on the left-hand side, it provides navigation among the various configuration/management objects accessible in iWD Manager.
- Details view (see page 137). 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.



Genesys intelligent Workload Distribution

Version: iWD-8.1.000.16

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

Name	Version	Owned by
Logging Service	iWD-8.1.000.16	SYSTEM
Database	iWD-8.1.000.16	SYSTEM
Interaction Server Connector	iWD-8.1.000.16	SYSTEM
Configuration Server Connector	iWD-8.1.000.16	SYSTEM
Workforce Management Connector	iWD-8.1.000.16	SYSTEM
Generic Capture Point	iWD-8.1.000.16	SYSTEM
Business Context Management Service	iWD-8.1.000.16	SYSTEM
Database Capture Point	iWD-8.1.000.16	SYSTEM
Kettle ETL Service	iWD-8.1.000.16	SYSTEM
Scheduled ETL Job	iWD-8.1.000.16	SYSTEM
Statistics Adapter	iWD-8.1.000.16	SYSTEM
Rules Service	iWD-8.1.000.16	SYSTEM
Scripting Service	iWD-8.1.000.16	SYSTEM
Webservice Capture Point	iWD-8.1.000.16	SYSTEM
XML File Capture Point	iWD-8.1.000.16	SYSTEM

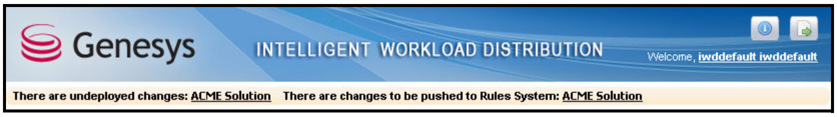
© 2012 Genesys Telecommunications Laboratories, Inc. All rights reserved.

Figure 27: iWD Manager User Interface

Header Bar

The iWD Manager Header bar (see Figure 28) 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 icon—Opens the iWD Manager Help.
- Logout—Logs the user out of the iWD Manager application.
- User—Displays the first and last name of the currently logged-in user.
- Notification of changes to be pushed to Genesys Rules System—Displays when there are changes to the current solution's business structure, such as the addition or deletion of a Solution, Department, or Process. The notification also includes a link to a screen that enables the user to push the changes to the Genesys Configuration Server, to make them available to the Genesys Rules System.


Genesys
INTELLIGENT WORKLOAD DISTRIBUTION
Welcome, iwddefault iwddefault

There are undeployed changes: ACME Solution There are changes to be pushed to Rules System: ACME Solution

Figure 28: iWD Manager Header Bar

Navigation Panel

The iWD Manager Navigation panel (see Figure 29) consists of the following UI objects:

- Tenant selection—On top of the navigation area; allows switching among iWD tenants. See “Tenant Selection” on page 135.
- Configuration sections—At bottom of the navigation area; allows switching among top-level configuration sections. See “Configuration Sections” on page 135.
- 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 136.
- Last Viewed—Between the navigation tree and its sections, provides quick access to the last-viewed panes. See “Last Viewed” on page 137.

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.

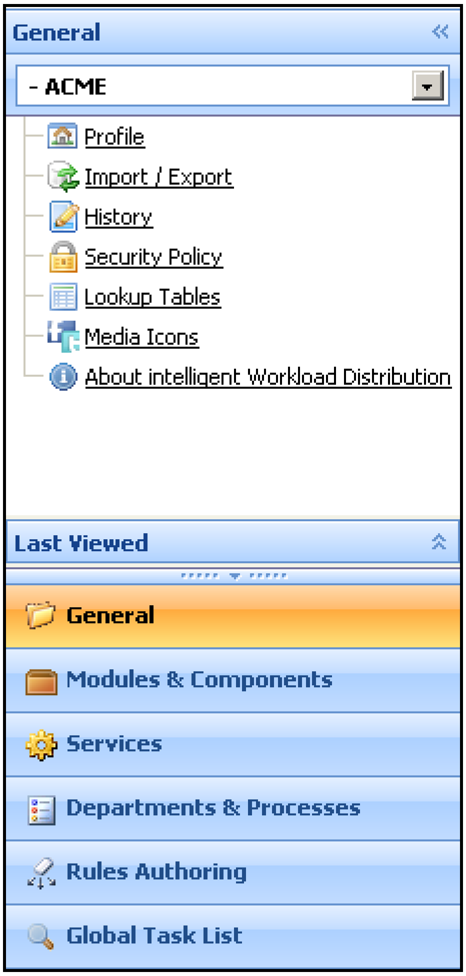


Figure 29: Navigation Panel

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. In a multi-tenant Configuration Server environment, where a single iWD managed tenant is configured, the tenant selector will default to that managed tenant after a user logs in to iWD Manager. See Figure 30.

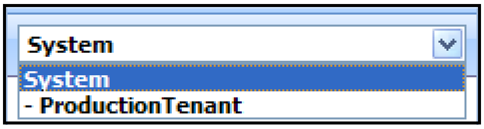


Figure 30: Tenant Selection

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 31 on page 136):

- General—General tenant configuration and actions, such as security policy and configuration import/export. For a detailed description, see “iWD Manager Overview” on page 127.
- Modules & Components—Modules and components that are owned by or accessible to the tenant. For a detailed description, see “Modules and Components” on page 162.
- Services—Solutions and services that are configured for a tenant. For a detailed description, see “Solutions and Services” on page 166.
- Departments & Processes—iWD business configuration, such as departments, processes, and rules. For a detailed description, see “Departments and Processes” on page 202.
- Rules Authoring—Launches the Genesys Rules Authoring Tool in a new browser tab or window (depending on browser configuration) and logs the user in automatically, navigating to the same tenant with which iWD Manager is working.
- Global Task List—iWD management views. For a detailed description, see Chapter 7, “Global Task List,” on page 249.

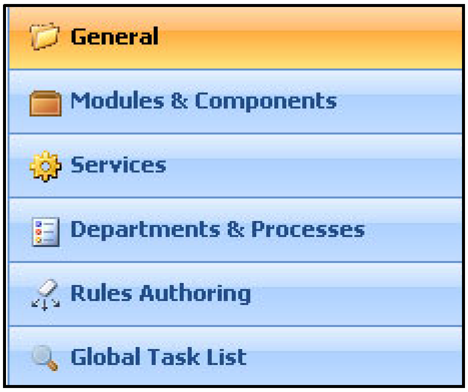


Figure 31: Configuration Sections

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 32

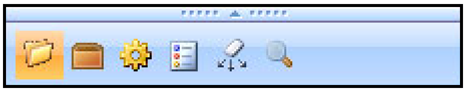


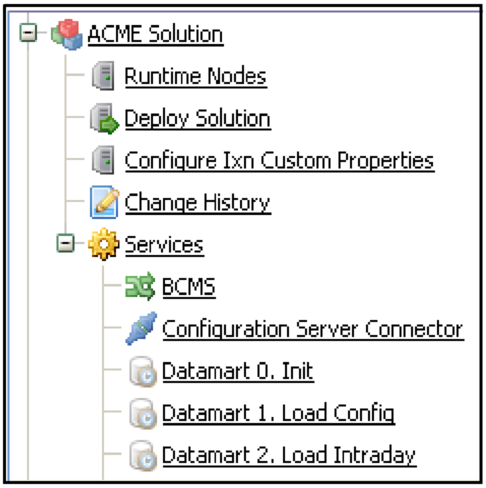
Figure 32: Compact Configuration Sections

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

Navigation Tree

The navigation tree, shown in Figure 33 on page 137, 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.

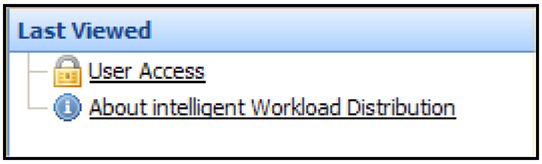


ACME Solution
Runtime Nodes
Deploy Solution
Configure Ixn Custom Properties
Change History
Services
BCMS
Configuration Server Connector
Datamart 0. Init
Datamart 1. Load Config
Datamart 2. Load Intraday

Figure 33: 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 34.



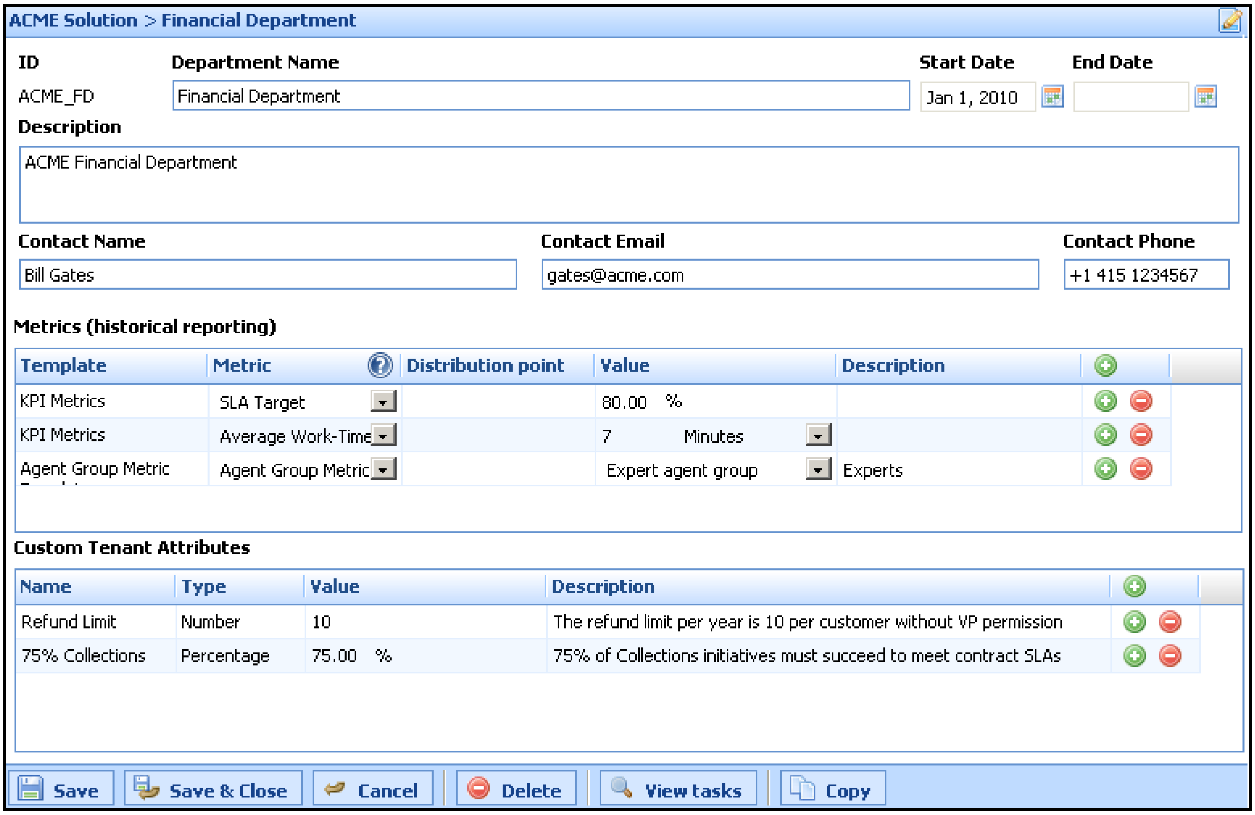
Last Viewed
User Access
About intelligent Workload Distribution

Figure 34: 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 134). Figure 35 illustrates an example of the Details view for a department.



ACME Solution > Financial Department

ID	Department Name	Start Date	End Date
ACME_FD	Financial Department	Jan 1, 2010	

Description

ACME Financial Department

Contact Name	Contact Email	Contact Phone
Bill Gates	gates@acme.com	+1 415 1234567

Metrics (historical reporting)

Template	Metric	Distribution point	Value	Description
KPI Metrics	SLA Target		80.00 %	
KPI Metrics	Average Work-Time		7 Minutes	
Agent Group Metric	Agent Group Metric		Expert agent group	Experts

Custom Tenant Attributes

Name	Type	Value	Description
Refund Limit	Number	10	The refund limit per year is 10 per customer without VP permission
75% Collections	Percentage	75.00 %	75% of Collections initiatives must succeed to meet contract SLAs

Figure 35: 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 36 on page 139):

- 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 and Processes. Services are pasted as new objects that must be saved. 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.

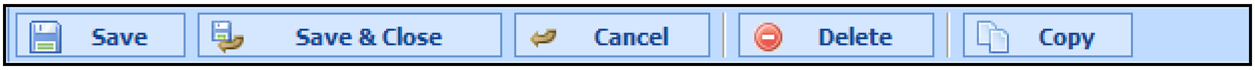


Figure 36: Details View Actions

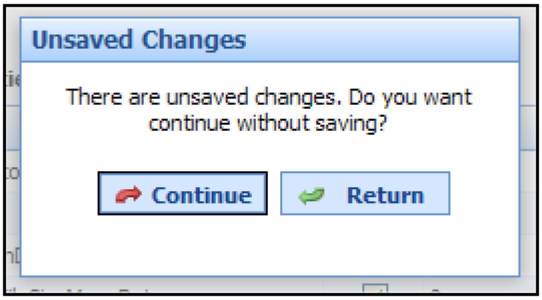
Note: The ability to modify an object also depends on the user's permissions. If the user does not have permission to modify a particular object, the previously described object actions will not be accessible.

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 37. 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,



There are unsaved changes. Do you want continue without saving?

Figure 37: Unsaved Changes

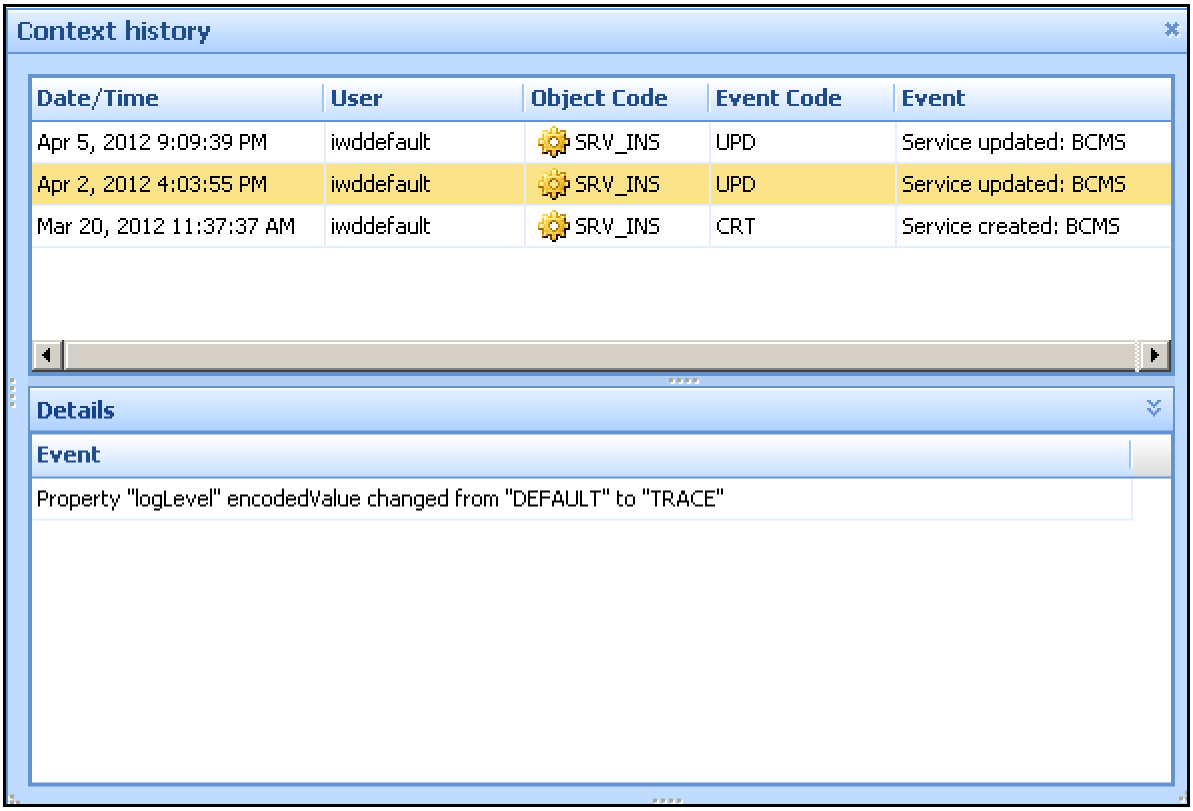
Context History

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 38 shows an example Context History for a Service.

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—The person who made the modification.
- Object Code—Description of the object type (such as `SRV_INS,` which means “Service Instance”).
- Event Code—Description of the action type (such as UPD, which means “Update”).

- Event—Formatted description of the change



Date/Time	User	Object Code	Event Code	Event
Apr 5, 2012 9:09:39 PM	iwddefault	SRV_INS	UPD	Service updated: BCMS
Apr 2, 2012 4:03:55 PM	iwddefault	SRV_INS	UPD	Service updated: BCMS
Mar 20, 2012 11:37:37 AM	iwddefault	SRV_INS	CRT	Service created: BCMS

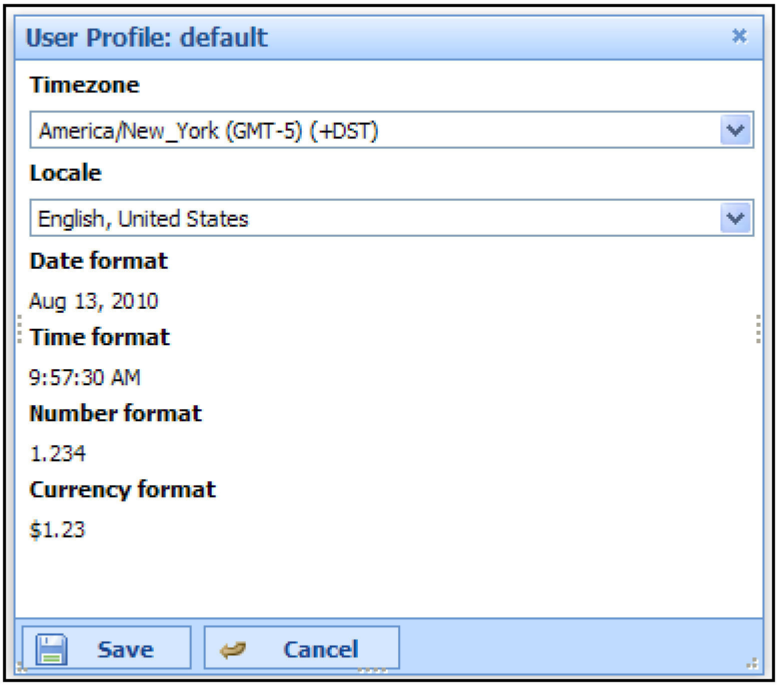
Event
Property "logLevel" encodedValue changed from "DEFAULT" to "TRACE"

Figure 38: Context History

Additionally, for most `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 39. To access the User Profile dialog box, click the user name in the Header bar (see “Header Bar” on page 133). 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.



User Profile: default

Timezone

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

Locale

English, United States

Date format

Aug 13, 2010

Time format

9:57:30 AM

Number format

1.234

Currency format

$1.23

Figure 39: User Profile Dialog Box

General Configuration

The General configuration section (see Figure 40) provides the ability to manage general tenant-level configuration:

- Use Profile view (see page 142) to view and modify general tenant attributes.
- Use Import and Export view (see page 144) to import and export tenant's configuration to and from XML files.
- Use Security Policies view (see page 146) to manage tenant roles and permissions.
- Use Managed Tenant Details view (see page 155) to create new managed tenants, and view and modify attributes of managed tenants.
- Use Lookup Tables view (see page 156) to specify tenant-specific constants as key/label pairs.
- Use Media Icons view ((see page 159) to manage the icons that are displayed in the Global Task List for each media type.
- Use History view (see page 160) to keep track of changes in tenant configuration.

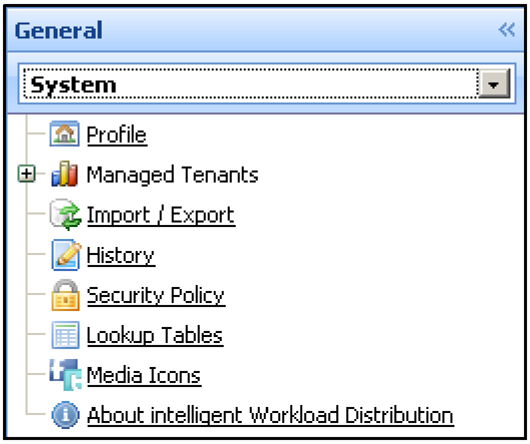
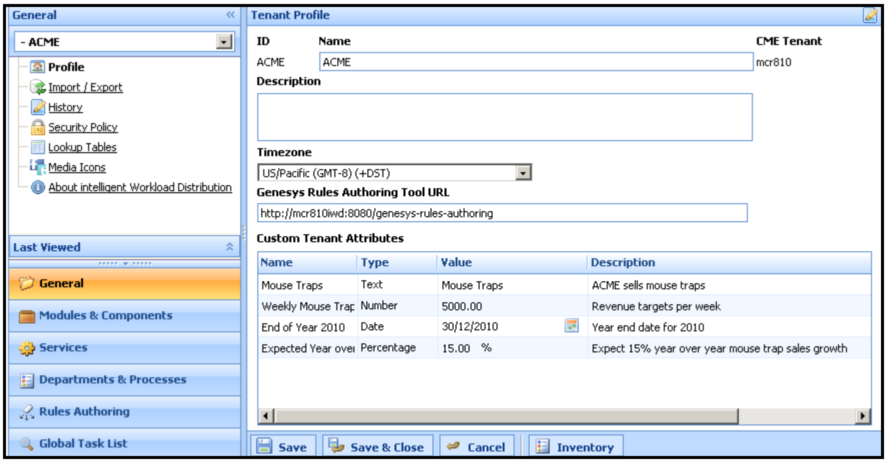


Figure 40: General Configuration Section

Profile

The Profile view (see Figure 41) displays and allows modification of general tenant details, such as name, description, and default time zone.



ID: ACME
Name: ACME
CME Tenant: mcr810
Description:
Timezone: US/Pacific (GMT-8) (+DST)
Genesys Rules Authoring Tool URL: http://mcr810iwd:8080/genesys-rules-authoring

Custom Tenant Attributes

Name	Type	Value	Description
Mouse Traps	Text	Mouse Traps	ACME sells mouse traps
Weekly Mouse Trap	Number	5000.00	Revenue targets per week
End of Year 2010	Date	30/12/2010	Year end date for 2010
Expected Year ovei	Percentage	15.00 %	Expect 15% year over year mouse trap sales growth

Figure 41: Profile View

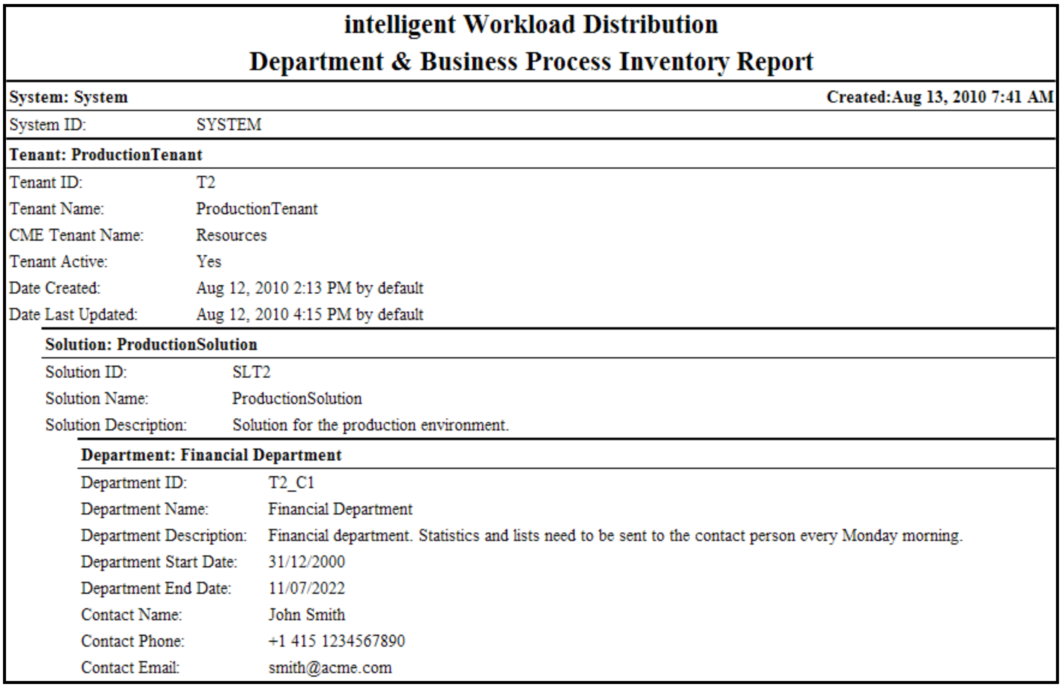
Table 3 lists the properties and actions available in the Profile view.

Table 3: 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 3: Profile View Attributes and Actions (Continued)

Property/Action	Description
CME Tenant	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. 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.
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.
Genesys Rules Authoring Tool URL	The Genesys Rules Authoring Tool URL can be launched from the iWD Manager. If the URL is not configured, the `Rules Authoring` node will not appear in the list of configuration sections. The URL must be in the following format: `http://<host>:<port>/genesys-rules-authoring` Where `<host>` and `<port>` are the host name and port number of the web application server where the Genesys Rules Authoring web application is deployed, and `genesys-rules-authoring` is the name of the Genesys Rules Authoring web application that is deployed on the web application server.
Custom Tenant Attributes	Custom tenant attributes that provide additional information about the tenant for reporting purposes. Refer to the *iWD 8.1 Data Mart Reference Guide* for more information.
Save/Save & Close/Cancel	Standard iWD Manager functions as described in “Object Actions” on page 138.
Inventory	Opens the tenant business process inventory report that contains detailed information about tenant business configuration (such as, departments and processes). See Figure 42 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 42: Inventory Report

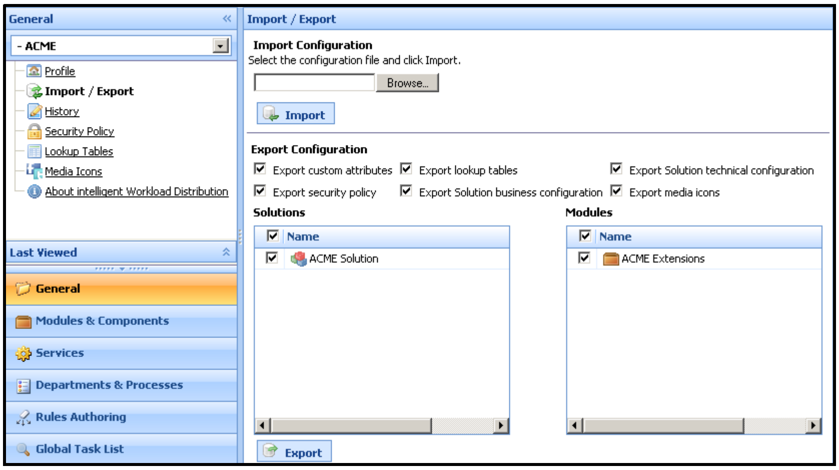
Import and Export

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

The Import/Export function will only import and export the iWD System and Tenant configuration. In most cases, you will also want to export and import the rule packages from the Genesys Rules System that correspond with your iWD Solution. For more information about the Import/Export function, see the related topics in the *Genesys Rules System 8.1 Rules Authoring Tool Help.*

Figure 43 shows an example of the Import/Export function.



Import Configuration

Select the configuration file and click Import.

Export Configuration

☑ Export custom attributes ☑ Export lookup tables ☑ Export Solution technical configuration

☑ Export security policy ☑ Export Solution business configuration ☑ Export media icons

Solutions

☑	Name
☑	ACME Solution

Modules

☑	Name
☑	ACME Extensions

Figure 43: Import/Export

Table 4 lists the properties and actions available in the Import/Export view.

Table 4: 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 142) in the exported configuration file.
Export security policy	Whether to include the tenant’s security policies (see page 146) in the exported configuration file.
Export lookup tables	Whether to include the tenant’s lookup tables (see page 156) 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.
Export Media Icons	Whether to include the tenant's media icons in the exported configuration file.
Solutions	Which of the tenant’s solutions (see page 167) to include in the exported configuration file.

Table 4: Import and Export Properties and Actions (Continued)

Property/Action	Description
Modules	Which of the tenant's modules (see page 163) to include in the exported configuration file.
Export	Exports the configuration objects that are selected above to an XML configuration file.

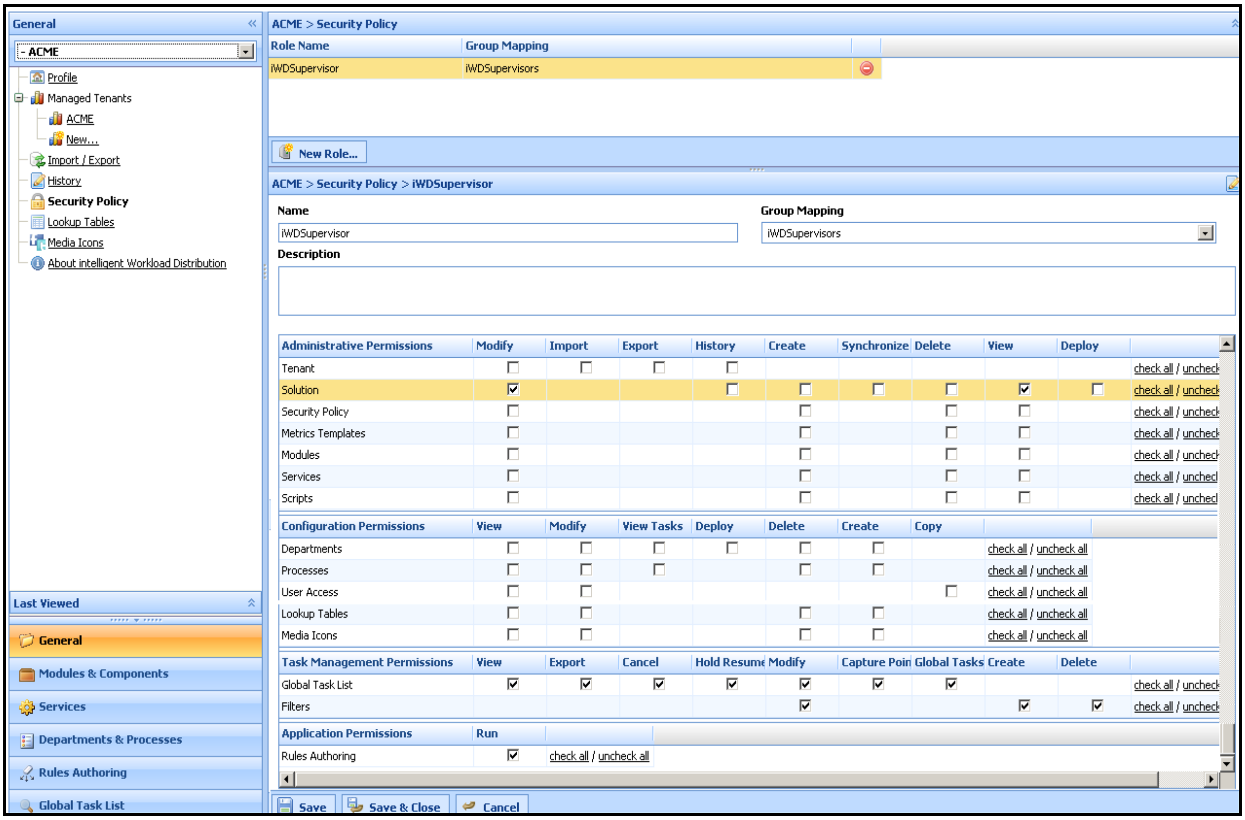
Security Policies

A Security Policy (see Figure 44) 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 the following four groups:

- Administrative permissions—Technical configuration (see Table 6 on page 148).
- Configuration permissions—Business configuration (see Table 7 on page 151).
- Task management permissions—The Global Task List view in iWD Manager (see Table 8 on page 153).
- Application permissions—The ability to launch the Genesys Rules Authoring Tool from iWD Manager (see Table 9 on page 154).

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, to which you choose to map 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.



Role Name	Group Mapping
iWDSupervisor	iWDSupervisors

Name: iWDSupervisor

Group Mapping: iWDSupervisors

Description:

Administrative Permissions	Modify	Import	Export	History	Create	Synchronize	Delete	View	Deploy	
Tenant	□	□	□	□						check all / uncheck
Solution	☑			□	□	□	□	☑	□	check all / uncheck
Security Policy	□				□		□	□		check all / uncheck
Metrics Templates	□				□		□	□		check all / uncheck
Modules	□				□		□	□		check all / uncheck
Services	□				□		□	□		check all / unchecl
Scripts	□				□		□	□		check all / unchecl

Configuration Permissions	View	Modify	View Tasks	Deploy	Delete	Create	Copy	
Departments	□	□	□	□	□	□		check all / uncheck all
Processes	□	□	□		□	□		check all / uncheck all
User Access	□	□					□	check all / uncheck all
Lookup Tables	□	□			□	□		check all / uncheck all
Media Icons	□	□			□	□		check all / uncheck all

Task Management Permissions	View	Export	Cancel	Hold Resume	Modify	Capture Poin	Global Tasks	Create	Delete	
Global Task List	☑	☑	☑	☑	☑	☑	☑			check all / uncheck
Filters					☑			☑	☑	check all / uncheck

Application Permissions	Run	
Rules Authoring	☑	check all / uncheck all

Figure 44: 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 5 lists the properties available for Security Policies.

Table 5: Security Policy Properties

Property	Description
Name	Role name. This is a mandatory field and must be unique within the tenant.
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 release 8.1, 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.

Genesys Rules System also supports role-based access control. The role permissions for the Genesys Rules Authoring Tool and Genesys Rules Development Tool are managed through Genesys Administrator. See the section “Role-Based Access Control", Chapter 5 in the *Genesys Rules System 8.1 Deployment Guide*.

Permission Groups

This section provides detailed descriptions of permissions that can be applied from each of the four permission groups: Administrative, Configuration, Task Management, and Application.

Administrative Permissions

Administrative permissions control the access that members in the security role have to technical configuration features in iWD Manager.

Administrative access control for each permission is described in Table 6.

Table 6: Administrative Permissions

Object	Permission	Access control
Tenant	Modify	Controls whether the role can edit and save properties of the iWD Tenant’s profile.
Tenant	Import	Controls whether the role can import XML configuration files into the iWD configuration database.
Tenant	Export	Controls whether the role can export XML configuration files from the iWD configuration database.
Tenant	History	Controls whether the role can view the context history of changes at the iWD Tenant level.
Solution	Modify	Controls whether the role can edit and save properties of iWD Solutions. This permission also requires the Solution View permission.

Table 6: Administrative Permissions (Continued)

Object	Permission	Access control
Solution	History	Controls whether the role can view the context history of changes at the iWD Solution level.
Solution	Create	Controls whether the role can create new iWD Solutions. This permission also requires the Solution View permission and the Solution Modify permission.
Solution	Synchronize	Controls whether the role can synchronize changes between the iWD Solution and the Genesys Configuration Server, for use by the Genesys Rules System. If this permission is not present, members of the role will not see the Push to Rules System link in the `Services > Solution` navigation tree. This permission also requires the Departments View permission under Configuration Permissions.
Solution	Delete	Controls whether the role can delete iWD Solutions. This permission also requires the Solution View permission.
Solution	View	Controls whether the role can view properties of iWD Solutions. This permission is normally combined with other permissions for which it is required.
Solution	Deploy	Controls whether the role can deploy iWD Solutions. This permission also requires the Solution View permission.
Security Policy	Modify	Controls whether the role can modify security policies. This permission also requires the Security Policy View permission.
Security Policy	Create	Controls whether the role can create new security policies. This permission also requires the Security Policy View permission.
Security Policy	Delete	Controls whether the role can delete security policies. This permission also requires the Security Policy View permission.
Security Policy	View	Controls whether the role can view security policies. This permission is normally combined with other permissions for which it is required.
Metrics Template	Modify	Controls whether the role can modify metrics templates. This permission also requires the Metrics Templates View permission.
Metrics Template	Create	Controls whether the role can create new metrics templates. This permission also requires the Metrics Templates View permission.
Metrics Template	Delete	Controls whether the role can delete metrics templates. This permission also requires the Metrics Templates View permission.

Table 6: Administrative Permissions (Continued)

Object	Permission	Access control
Metrics Template	View	Controls whether the role can view metrics templates. This permission is normally combined with other permissions for which it is required.
Modules	Modify	Controls whether the role can modify properties of modules. This permission also requires the Modules View permission.
Modules	Create	Controls whether the role can create new modules. This permission also requires the Modules View permission.
Modules	Delete	Controls whether the role can delete modules. This permission also requires the Modules View permission.
Modules	View	Controls whether the role can view modules. This permission is normally combined with other permissions for which it is required.
Services	Modify	Controls whether the role can modify properties of services. This permission also requires the Services View permission.
Services	Create	Controls whether the role can create new services. This permission also requires the Services View permission.
Services	Delete	Controls whether the role can delete services. This permission also requires the Services View permission.
Services	View	Controls whether the role can view services. This permission is normally combined with other permissions for which it is required.
Scripts	Modify	Controls whether the role can modify transformation scripts. This permission also requires the Scripts View permission.
Scripts	Create	Controls whether the role can create new transformation scripts. This permission also requires the Scripts View permission.
Scripts	Delete	Controls whether the role can delete transformation scripts. This permission also requires the Scripts View permission.
Scripts	View	Controls whether the role can view transformation scripts. This permission is normally combined with other permissions for which it is required.

Configuration Permissions

Configuration permissions control the access that members in the security role have to business configuration features in iWD Manager.

Configuration access control for each permission is described in the Table 7.

Table 7: Configuration Permissions

Object	Permission	Access control
Department	View	Controls whether the role can view Departments. This permission is normally combined with other permissions for which it is required.
Department	Modify	Controls whether the role can modify properties of Departments. This permission also requires the Departments View permission.
Department	View Tasks	Controls whether the role can view tasks at the Department level, through the Global Task List. This permission also requires the Global Task List View permission. It also requires that the role has access to one or more Departments through the `Departments & Processes > User Access` screen.
Department	Deploy	Controls whether the role can deploy an iWD Solution from within the Departments & Processes view. **Note:** The permission controls access to the same capability as the Solution>Deploy permission. This permission also requires the Departments View permission.
Department	Delete	Controls whether the role can delete Departments. This permission also requires the Departments View permission.
Department	Create	Controls whether the role can create new Departments. This permission also requires the Departments View permission.
Process	View	Controls whether the role can view Processes. This permission is normally combined with other permissions for which it is required.
Process	Modify	Controls whether the role can modify properties of Processes. This permission also requires the Processes View permission.
Process	View Tasks	Controls whether the role can view tasks at the Process level, through the Global Task List. This permission also requires the Global Task List View permission and the Departments View Tasks permission. It also requires that the role has access to one or more Processes through the `Departments & Processes > User Access` screen.
Process	Delete	Controls whether the role can delete Processes. This permission also requires the Processes View permission.
Process	Create	Controls whether the role can create new Processes. This permission also requires the Processes View permission.

Table 7: Configuration Permissions (Continued)

Object	Permission	Access control
User Access	View	Controls whether the role can view the User Access screen. This permission is normally combined with other permissions that require it. This permission also requires the Departments View permission.
User Access	Modify	Controls whether the role can modify User Access properties (the assignment of roles to view Departments and Processes). This permission also requires the User Access View permission and the Departments View permission.
User Access	Copy	Controls whether the role can copy user access between Solutions. This permission also requires the User Access View permission and the Departments View permission.
Lookup Tables	View	Controls whether the role can view Lookup Tables.
Lookup Tables	Modify	Controls whether the role can modify properties of Lookup Tables. This permission also requires the Lookup Tables View permission.
Lookup Tables	Delete	Controls whether the role can delete Lookup Tables. This permission also requires the Lookup Tables View permission.
Lookup Tables	Create	Controls whether the role can create new Lookup Tables. This permission also requires the Lookup Tables View permission.
Media Icons	Modify	Controls whether the role can modify mapping of media types to media icons. This permission also requires the Media Icons View permission.
Media Icons	Delete	Controls whether the role can create new mappings of media types to media icons. This permission also requires the Media Icons View permission.
Media Icons	Create	Controls whether the role can delete the mappings of media types to media icons. This permission also requires the Media Icons View permission.
Media Icons	View	Controls whether the role can view the mappings of media types to media icons. This permission is normally combined with other permissions for which it is required.

Task-Management Permissions

Task-management permissions control the access that members in the security role have to task-management features in iWD Manager.

Task-management access control for each permission is described in the Table 8.

Table 8: Task-Management Permissions

Object	Permission	Access control
Global Task List	View	Controls whether the role can view tasks and filters in the Global Task List. It must be combined with Departments View Tasks or Processes View Tasks, with the appropriate User Access configured, or it may be combined with Global Task List Capture Points or Global Task List Global Tasks permissions, or some combination of these. If no additional permissions are applied with Global Task List View, there will be no access to any iWD business structure level(s) at which tasks can be viewed.
Global Task List	Export	Controls whether the role can export tasks to an XML file. It must be combined with Global Task List View, together with the other prerequisites described for that permission.
Global Task List	Cancel	Controls whether the role can cancel tasks from the Global Task List. It must be combined with Global Task List View, together with the other prerequisites described for that permission.
Global Task List	Hold/Resume	Controls whether the role can hold/resume tasks from the Global Task List. It must be combined with Global Task List View, together with the other prerequisites described for that permission.
Global Task List	Modify	Controls whether the role can modify attributes of tasks from the Global Task List, including the ability to `Save and Restart` the processing of the task. It must be combined with Global Task List View, together with the other prerequisites described for that permission.
Global Task List	Capture Points	Controls whether the role can view tasks at the Capture Point level. It must be combined with the Global Task List View permission.
Global Task List	Global Tasks	Controls whether the role can view tasks at the Solution level. It must be combined with the Global Task List View permission.
Filters	Modify controls	Controls whether the role can modify both Private and Public filters. The role can modify any property of the filter, including changing the filter's scope from Private to Public or Public to Private. It must be combined with the Global Task List View permission.

Table 8: Task-Management Permissions (Continued)

Object	Permission	Access control
Filters	Create	Controls whether the role can create both Private and Public filters. It must be combined with the Global Task List View permission.
Filters	Delete	Controls whether the role can delete both Private and Public filters. It must be combined with the Global Task List View permission.

Application Permissions

Application permissions control the access that members in the security role have to the Rules Authoring link in the iWD Manager navigation menu.

Application access control for each permission is described in the Table 9.

Table 9: Application Permissions

Object	Permission	Access control
Rules Authoring	Run	Controls whether the role has access to the Rules Authoring link in the iWD Manager navigation menu. This permission also requires that the property Genesys Rules Authoring Tool URL be configured on the `General > Profile` screen of the Tenant.

Important Information About Filtering Permissions

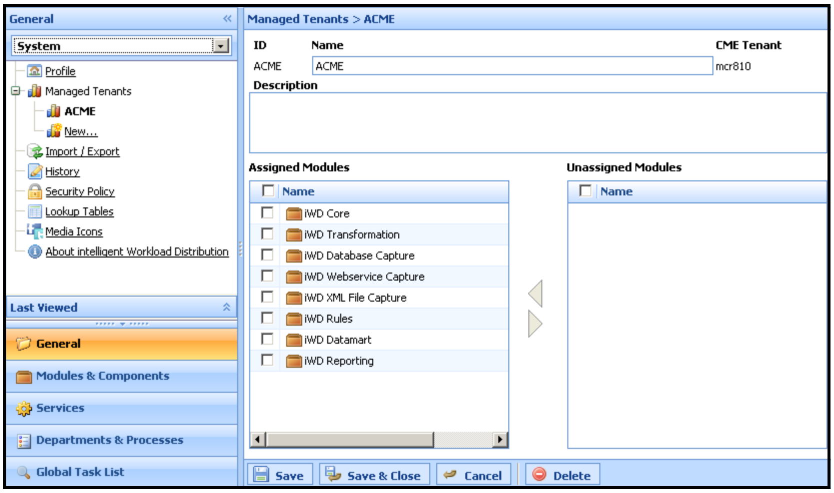
The following additional information is also important to note when you are filtering permissions:

- Public filters can be changed from Private by any user that has Modify filter permissions.
- The owner of a Private filter can always modify or delete that filter, even if they do not have Delete and/or Modify security role permissions.
- The username of the owner of a Private filter is displayed on the `Filter` configuration screen to make referencing easy..

Note: When security roles from previous versions are migrated to iWD 8.1.0, none of the above mentioned permissions are enabled, by default.

Managed Tenant Details

The Managed Tenant Details view (see Figure 45 on page 155) displays and enables you to modify general managed tenant attributes (see “Tenants” on page 45).



ID	Name	CME Tenant
ACME	ACME	mcr810

Description

Assigned Modules

□ Name
□ iWD Core
□ iWD Transformation
□ iWD Database Capture
□ iWD Webservice Capture
□ iWD XML File Capture
□ iWD Rules
□ iWD Datamart
□ iWD Reporting

Unassigned Modules

□ Name

Figure 45: Managed Tenant

Table 10 lists the attributes and actions available in Managed Tenant view.

Table 10: 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	The name of the corresponding tenant in Genesys Configuration Server. 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. In a single-tenant environment the configured managed tenant must map to the `Resources` tenant in Genesys Configuration Server. An iWD Managed Tenant has a one-to-one relationship with a tenant in Genesys Configuration Server.
Description	Tenant description.

Table 10: Managed Tenant Properties and Actions (Continued)

Property/Action	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 138.

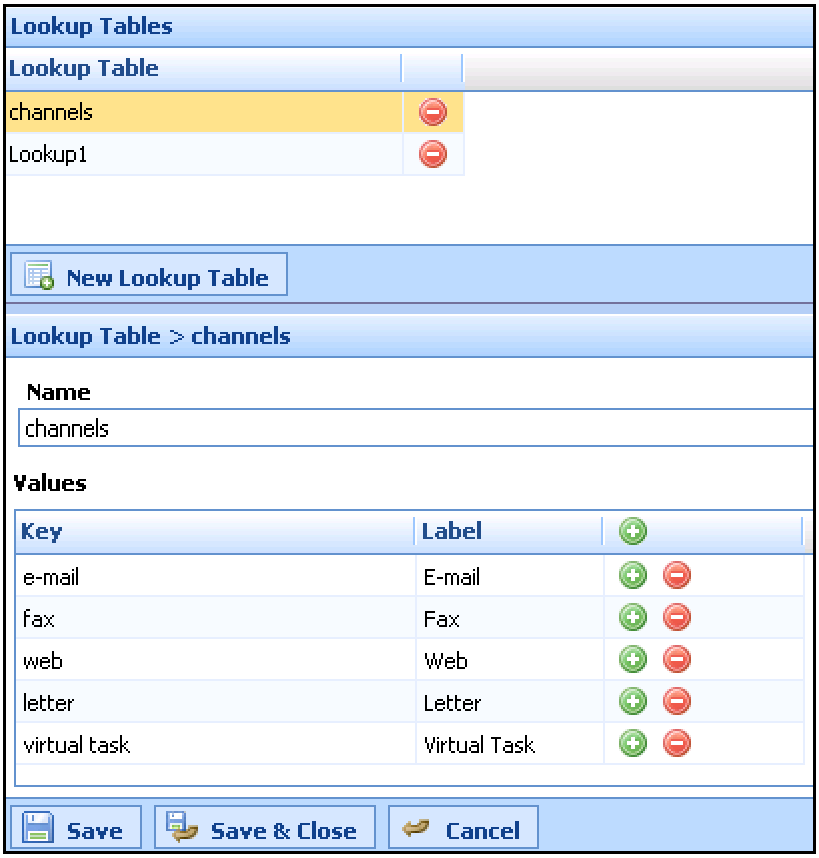
Lookup Tables

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 46.

Although business rules are managed in the Genesys Rules System in iWD 8.1, it is still possible to create rule parameters that use values from iWD Lookup Tables. For an example of how to configure this, see the `taskChannels` parameter in the iWD Standard Rules Template.

The `taskChannels` parameter is configured as a `database` type rule parameter. The configuration of that parameter instructs the Genesys Rules Authoring Tool how to query the iWD configuration database to retrieve the values of the out-of-the-box iWD Lookup Table called `channels`, as shown in Figure 45.

To create additional rule parameters that will retrieve the values from other Lookup Tables, you can make copies of the `taskChannels` parameter and modify the SQL query, changing the name of the Lookup Table from `channels` to the name of your Lookup Table.



Lookup Table
channels
Lookup1

Name
channels

Values

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

Figure 46: Lookup Tables

Distribution Points and Lookup Tables

The concept of Distribution Points has changed from iWD 7.6.1 to 8.0. Distribution Points are no longer required. Instead, in 8.1 (and 8.0), all tasks are immediately distributed to Interaction Server a task is captured. To retain the reporting functionality for customers that are upgrading from iWD 7.6.1, Distribution Points remain as attributes that can be assigned to tasks in business rules, and there is still a Distribution Point dimension in the iWD Data Mart schema.

In iWD 8.1, Distribution Points must be configured as Lookup Tables (see page 156) at the `Tenant` level. The following procedure describes the steps that are used to configure Distribution Points as Lookup Tables.

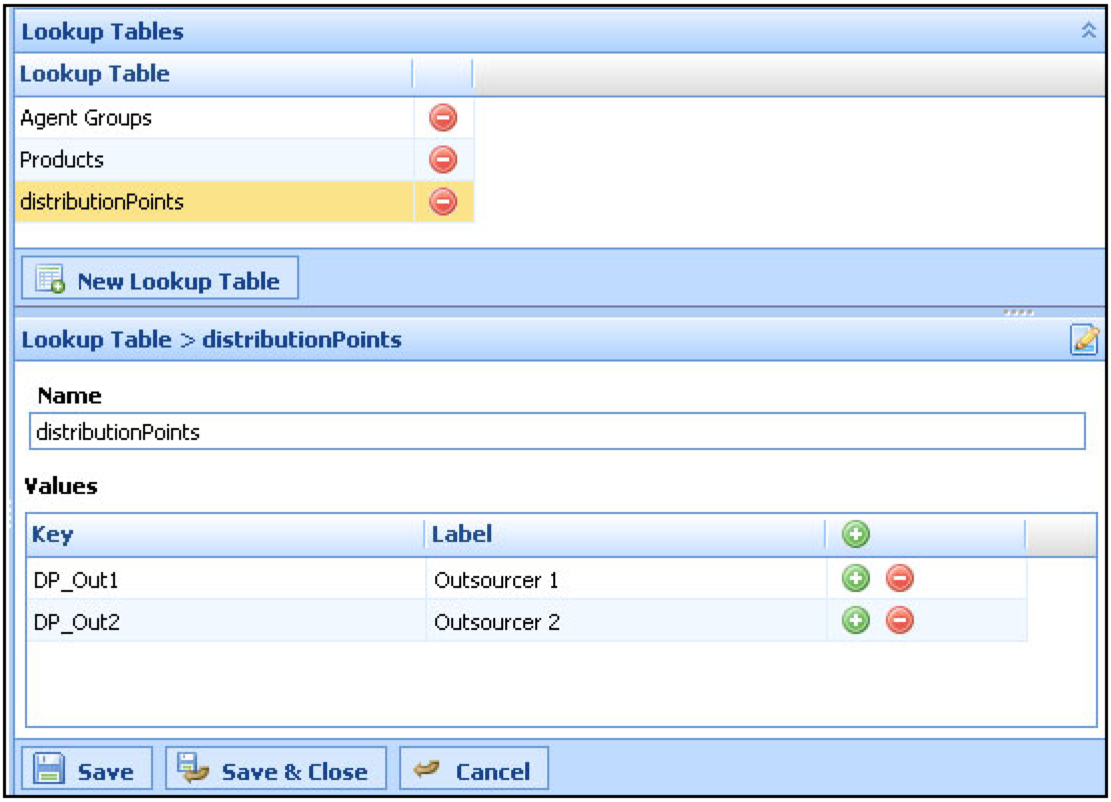
Procedure:
Configuring Distribution Points as Lookup Tables

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 47. 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.



Lookup Table
Agent Groups
Products
distributionPoints

Lookup Table > distributionPoints

Name: distributionPoints

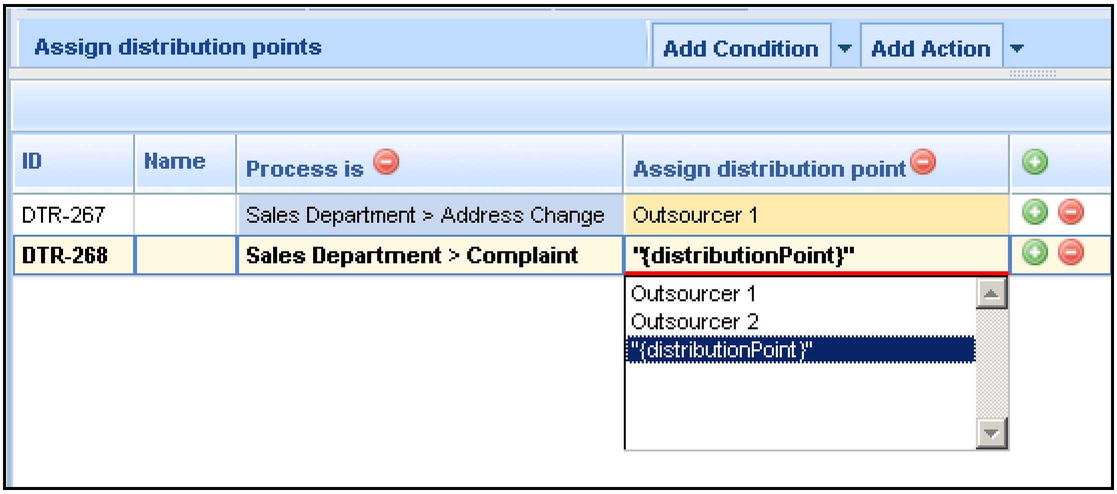
Values

Key	Label
DP_Out1	Outsourcer 1
DP_Out2	Outsourcer 2

Figure 47: Create a Distribution Point Lookup Table

2. Start using the rule action `Assign distribution point` in your business rules.

   There is a rule action in the iWD Standard Rule Template called `Assign distribution point` that uses a rule parameter configured to read the values from the `distributionPoints` Lookup Table. This rule action can be used in a business rule. See Figure 48.



ID	Name	Process is	Assign distribution point
DTR-267		Sales Department > Address Change	Outsourcer 1
DTR-268		Sales Department > Complaint	"{distributionPoint}"

Outsourcer 1
Outsourcer 2
"{distributionPoint}"

Figure 48: Assign Distribution Point in Rule

End of procedure

Media Icons

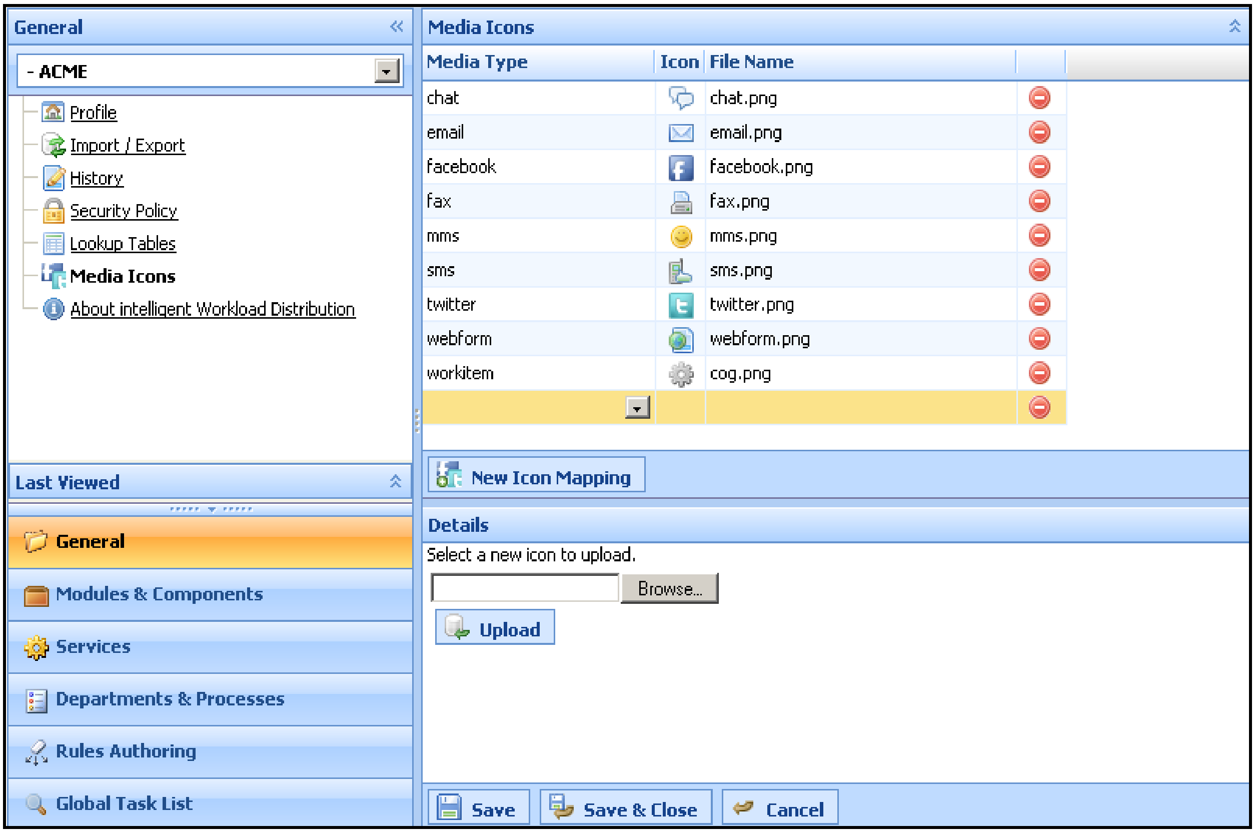
The Media Icons page (see Figure 49) provides a way to map graphical icons to media types to display in the iWD Global Task List. iWD comes with some preloaded icons, but additional icons can be uploaded and mapped to existing media types, or new media types. Icons must be 16x16, but can be in any file format, such as `.gif`, `.jpg`, or `.bmp`.

Table 11: Media Icons View Properties and Actions

Property/Action	Description
Media Type	The media type to which the icon will be mapped. This list of media types is retrieved dynamically from Genesys Configuration Server. Media types are a type of Business Attribute in Configuration Server. Genesys provides out-of-the-box media types but new custom media types can be added.
Icon	The graphical 16x16 icon that will be displayed in the Global Task List when a task is of the media type to which the icon is mapped.
File Name	The file name of the icon.
New Icon Mapping	The function that provides a way for the user to upload a new icon from the file system.
Browse	The function that opens a `File Upload` dialog window to enable the user to browse the file system to select an icon.

Table 11: Media Icons View Properties and Actions (Continued)

Property/Action	Description
Upload	The function that uploads the icon that was previously selected by the user through the Browse action.
Save/Save and Close/Cancel	The standard iWD Manager functions, as described in “Object Actions” on page 138.



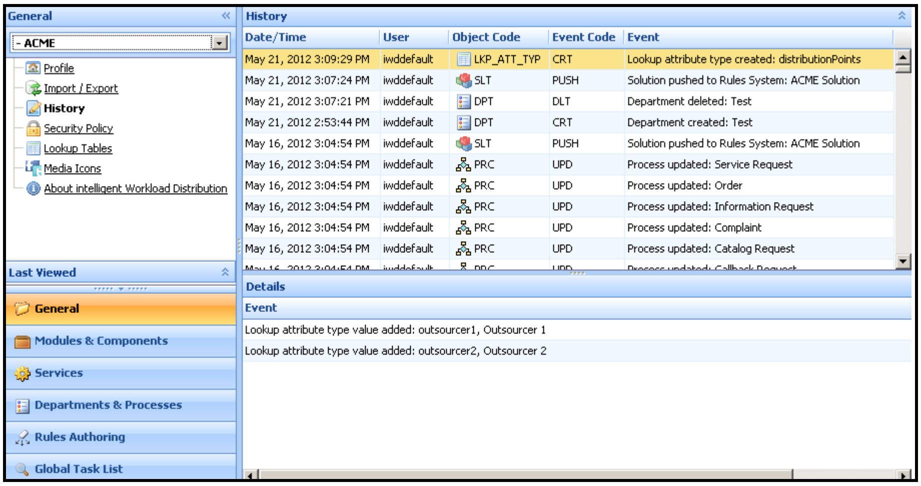
Media Type	Icon	File Name
chat		chat.png
email		email.png
facebook		facebook.png
fax		fax.png
mms		mms.png
sms		sms.png
twitter		twitter.png
webform		webform.png
workitem		cog.png

Select a new icon to upload.

Figure 49: Media Icons

History

The History page (see Figure 50) 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.



Date/Time	User	Object Code	Event Code	Event
May 21, 2012 3:09:29 PM	iwddefault	LKP_ATT_TYP	CRT	Lookup attribute type created: distributionPoints
May 21, 2012 3:07:24 PM	iwddefault	SLT	PUSH	Solution pushed to Rules System: ACME Solution
May 21, 2012 3:07:21 PM	iwddefault	DPT	DLT	Department deleted: Test
May 21, 2012 2:53:44 PM	iwddefault	DPT	CRT	Department created: Test
May 16, 2012 3:04:54 PM	iwddefault	SLT	PUSH	Solution pushed to Rules System: ACME Solution
May 16, 2012 3:04:54 PM	iwddefault	PRC	UPD	Process updated: Service Request
May 16, 2012 3:04:54 PM	iwddefault	PRC	UPD	Process updated: Order
May 16, 2012 3:04:54 PM	iwddefault	PRC	UPD	Process updated: Information Request
May 16, 2012 3:04:54 PM	iwddefault	PRC	UPD	Process updated: Complaint
May 16, 2012 3:04:54 PM	iwddefault	PRC	UPD	Process updated: Catalog Request
May 16, 2012 3:04:54 PM	iwddefault	PRC	UPD	Process updated: Callback Request

Event
Lookup attribute type value added: outsourcer1, Outsourcer 1
Lookup attribute type value added: outsourcer2, Outsourcer 2

Figure 50: History

Table 12 lists the properties and actions are available in History view.

Table 12: 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: • `TNT`: Tenant • `SLT`: Solution • `SRV_INS`: Service Instance • `RNT_ND`: Runtime Node • `MDL`: Module • `PRC`: Process • `DPT`: Department • `ROLE`: Security Policy Role
Event Code	Represents the type of activity: • `CRT`: Create • `UPD`: Update • `DLT`: Delete • `IMP`: Import • `EXP`: Export

Table 12: History View Properties and Actions (Continued)

Property/Action	Description
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 169) that are based on service templates, however, are configurable.
  - Metric templates (see page 164)
  - Scripts—These are transformation scripts that are only used with the legacy iWD capture point services (see Appendix B on page 277).
- A module (see page 163) is a group of components.

Note: Starting in iWD 8.1, rule templates are no longer configured in iWD Manager. They are created in the Genesys Rules Development Tool.

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.

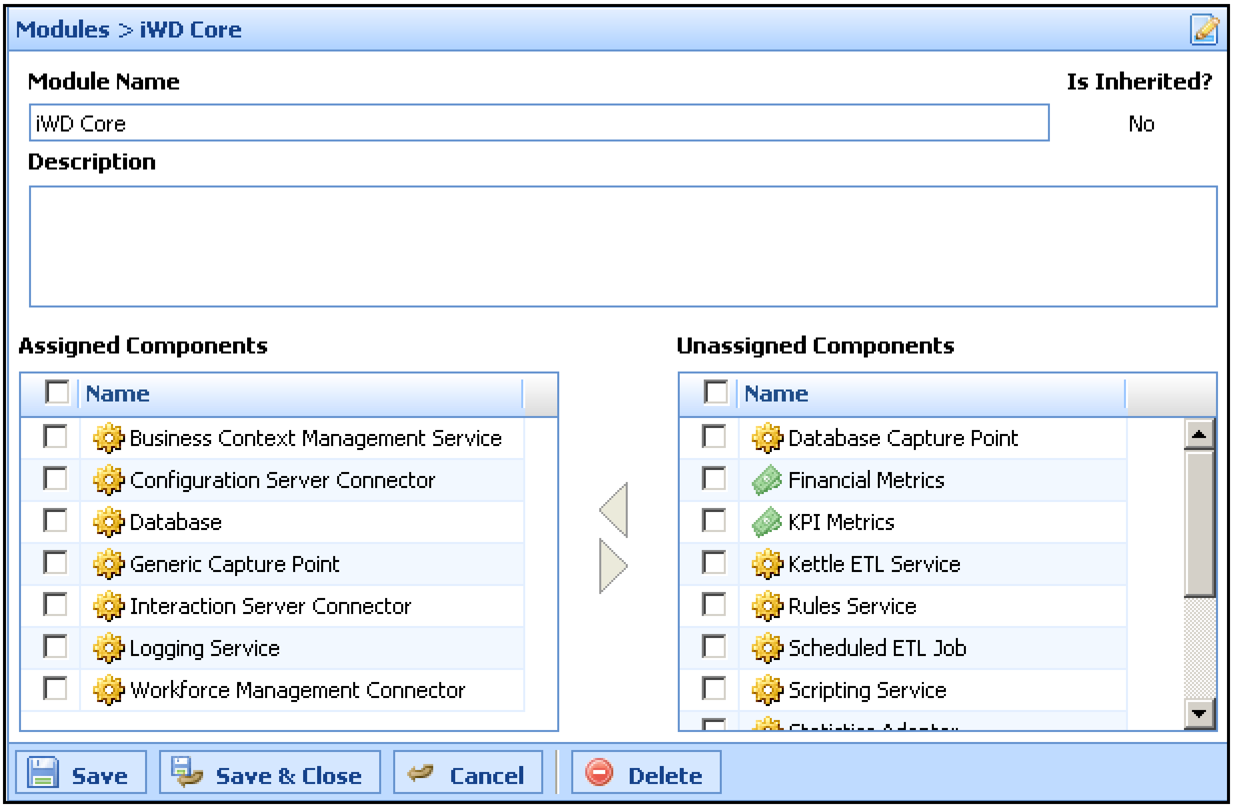
Modules

Modules bundle a set of iWD components into a named functional area. Modules are used to group *features*, such as reporting. Each feature can be implemented by one or more components where the component can be a service, 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 51 shows an exampe.



Modules > iWD Core

Module Name: iWD Core

Is Inherited? No

Description:

Assigned Components

□	Name
□	Business Context Management Service
□	Configuration Server Connector
□	Database
□	Generic Capture Point
□	Interaction Server Connector
□	Logging Service
□	Workforce Management Connector

Unassigned Components

□	Name
□	Database Capture Point
□	Financial Metrics
□	KPI Metrics
□	Kettle ETL Service
□	Rules Service
□	Scheduled ETL Job
□	Scripting Service
□	Statistics Adapter

Figure 51: Module

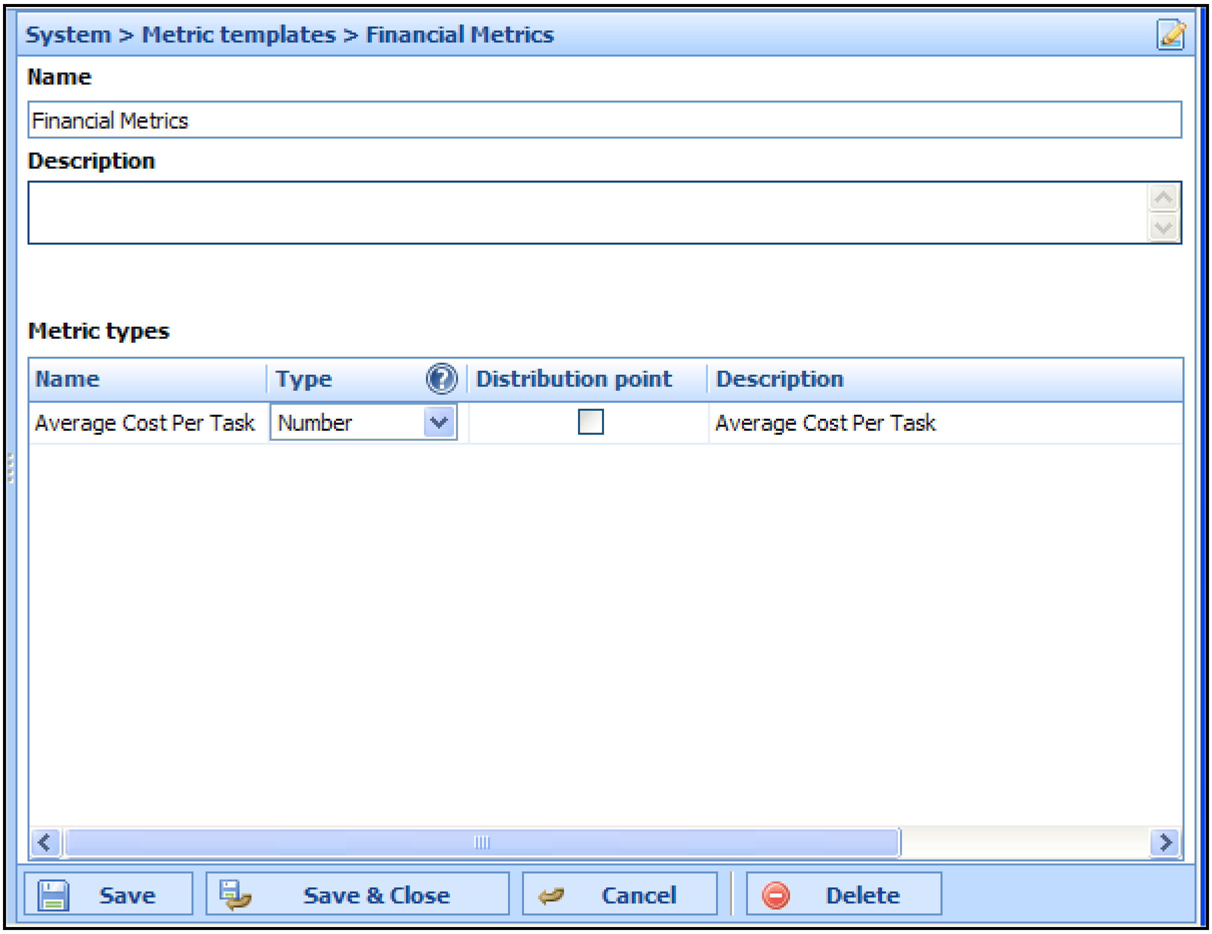
Table 13 lists the properties and actions available in the Modules view:

Table 13: 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, metrics templates, and scripts that belong to the module.
Unassigned Components	Services templates that do not belong to the module.
Save/Save & Close/ Cancel/Delete	Standard iWD Manager functions as described in "Object Actions" on page 138.

Metric Templates

Metric 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. Metric templates are just like any other module components and must be assigned to a module to be available under departments and processes. Figure 52 shows an example.



System > Metric templates > Financial Metrics

Name: Financial Metrics

Description:

Metric types

Name	Type	Distribution point	Description
Average Cost Per Task	Number	□	Average Cost Per Task

Figure 52: MetricTemplate

Table 14 lists value types for the Metric template. For more information about Metrics, see “Metrics” on page 206.

Table 14: 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.
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

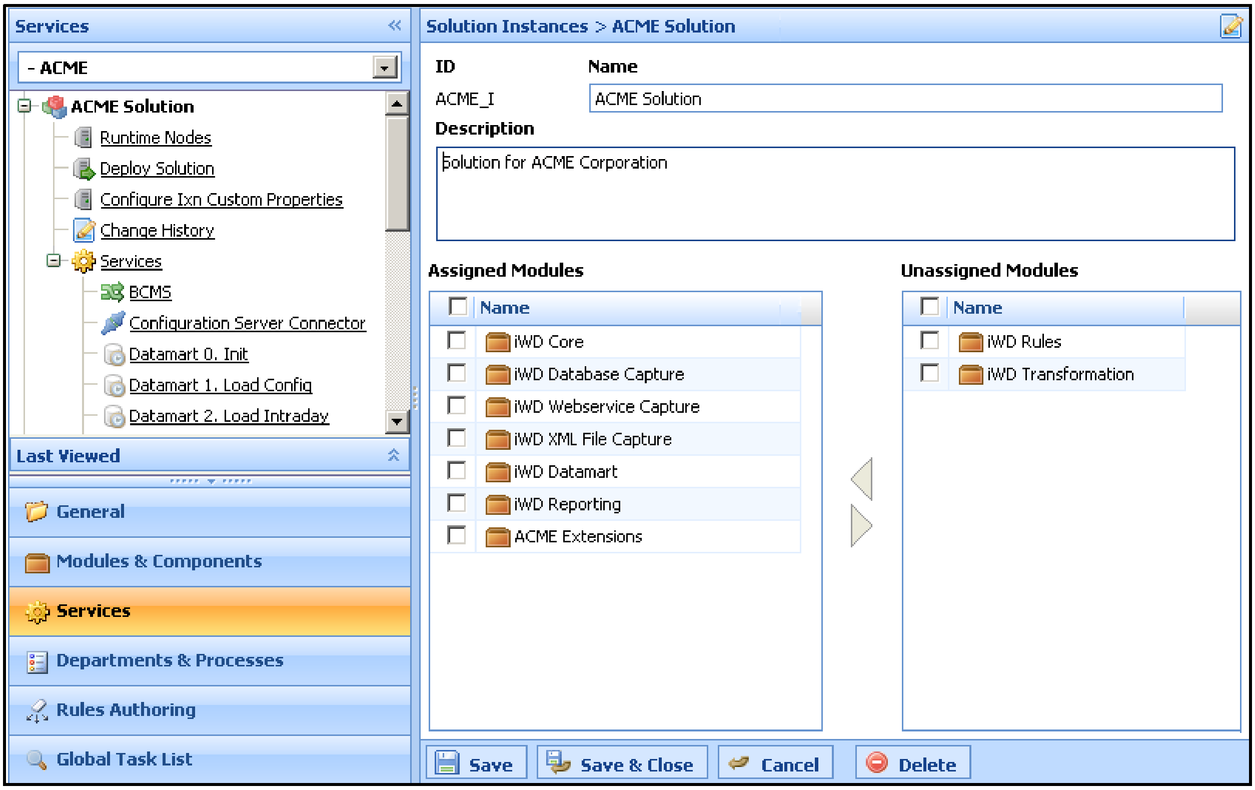
The Services configuration section allows the setup of iWD runtime instances for a specific infrastructure and environment:

- Use Solution Details view (see page 167) to create new solutions, as well as to view and modify general solution attributes.
- Use Runtime Nodes view (see page 168) to define runtime nodes in which services will run.
- Use Services view (see page 169) to monitor and manage service status.
- Use Service Details view (see page 171) to create new services, as well as to view and modify configuration of existing services.
- Use Deployment view (see page 173) to deploy solution configuration to runtime nodes, as well as to activate services.
- Use Configure Custom Ixn Properties view (see page 175) to configure Interaction Server databases and Configuration Server application options for iWD.
- Use Change History view (see page 176) 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 46 of Chapter 3. Figure 53 shows an example.



ID: ACME_I

Name: ACME Solution

Description: Solution for ACME Corporation

Assigned Modules

□	Name
□	iWD Core
□	iWD Database Capture
□	iWD Webservice Capture
□	iWD XML File Capture
□	iWD Datamart
□	iWD Reporting
□	ACME Extensions

Unassigned Modules

□	Name
□	iWD Rules
□	iWD Transformation

Figure 53: Solution Details

Table 15 lists the properties and actions available in Solution Details view.

Table 15: Solution Details Attributes and Actions

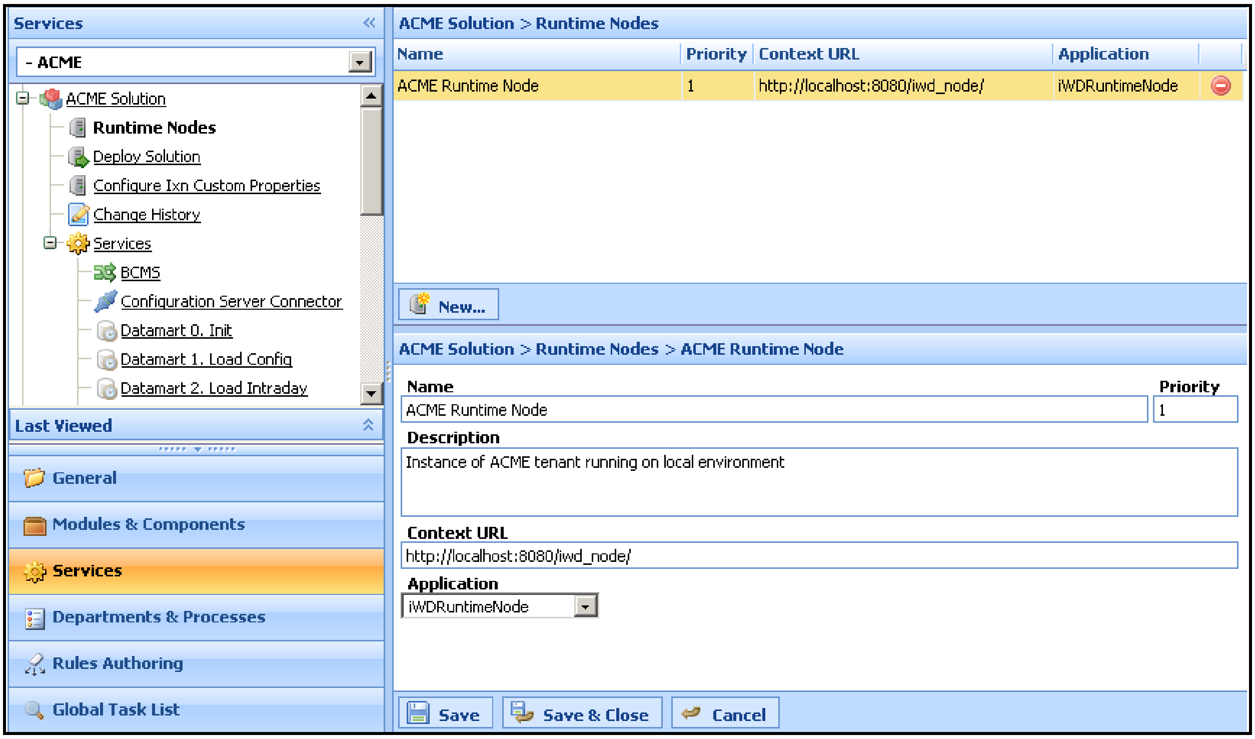
Property/Action	Description
ID	The ID of the solution. This field becomes read-only after the solution is created.
Name	The name of the solution.
Description	A description of the solution.
Assigned Modules	Modules (or functionality) that are assigned to the solution. Components such as Services and Metric templates from these modules are available for use in the solution.

Table 15: Solution Details Attributes and Actions (Continued)

Property/Action	Description
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 138.

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 47 for more information about runtime nodes. Figure 54 shows an example.



Name	Priority	Context URL	Application
ACME Runtime Node	1	http://localhost:8080/iwd_node/	iWDRuntimeNode

Name: ACME Runtime Node
Priority: 1
Description: Instance of ACME tenant running on local environment
Context URL: http://localhost:8080/iwd_node/
Application: iWDRuntimeNode

Figure 54: Runtime Nodes

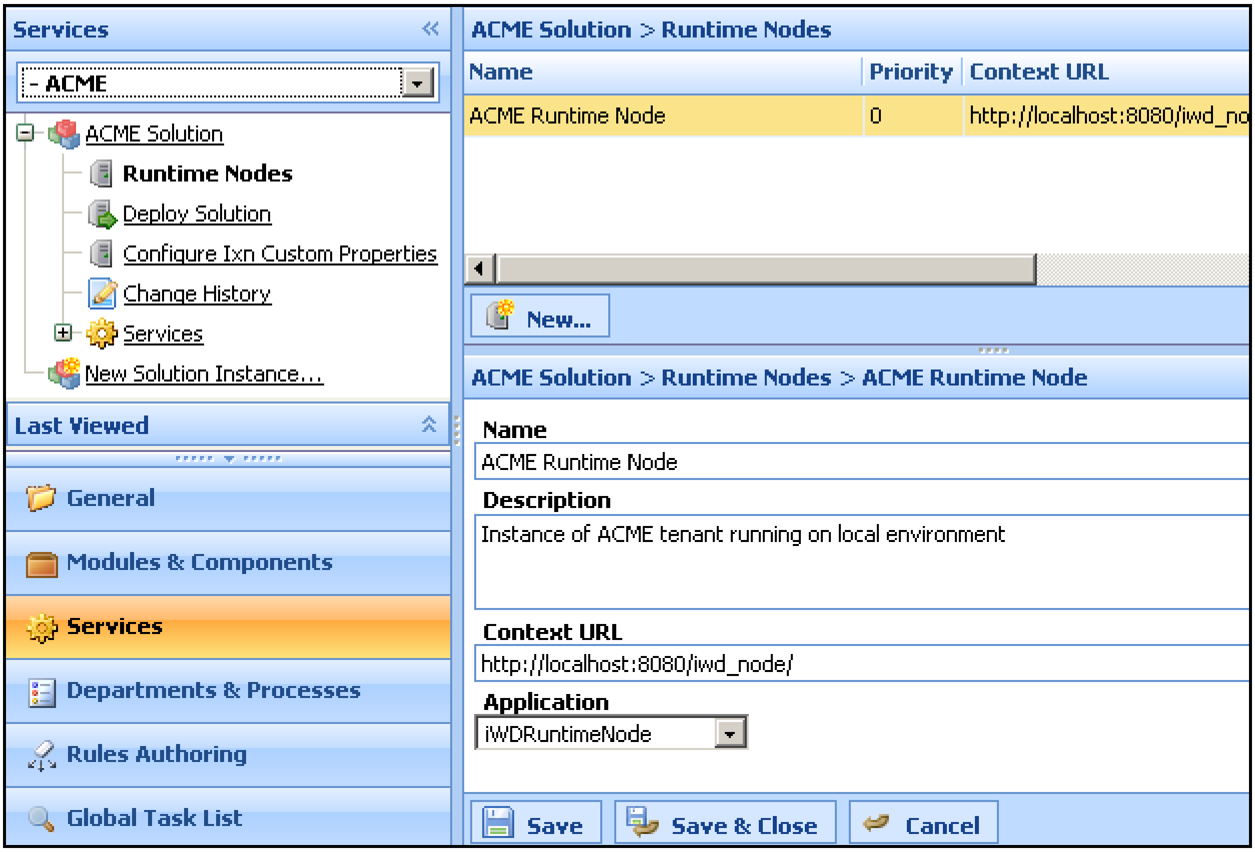
Table 16 describes the properties and actions that are available in Runtime Nodes view.

Table 16: 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 `http://localhost:8080/iwd_node`, where `8080` is the port used by Tomcat, and `iwd_node` is the Runtime Node webapp in the Tomcat container.
New...	Allows for definition of a new runtime node.
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 138.

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 47 of Chapter 3 for more information about services. Figure 55 shows an example.



Name	Priority	Context URL
ACME Runtime Node	0	http://localhost:8080/iwd_no

Name
ACME Runtime Node

Description
Instance of ACME tenant running on local environment

Context URL
http://localhost:8080/iwd_node/

Application
iWDRuntimeNode

Figure 55: Runtime Nodes

Table 17 describes the properties and actions that are available in the Services view.

Table 17: Services View Properties and Actions

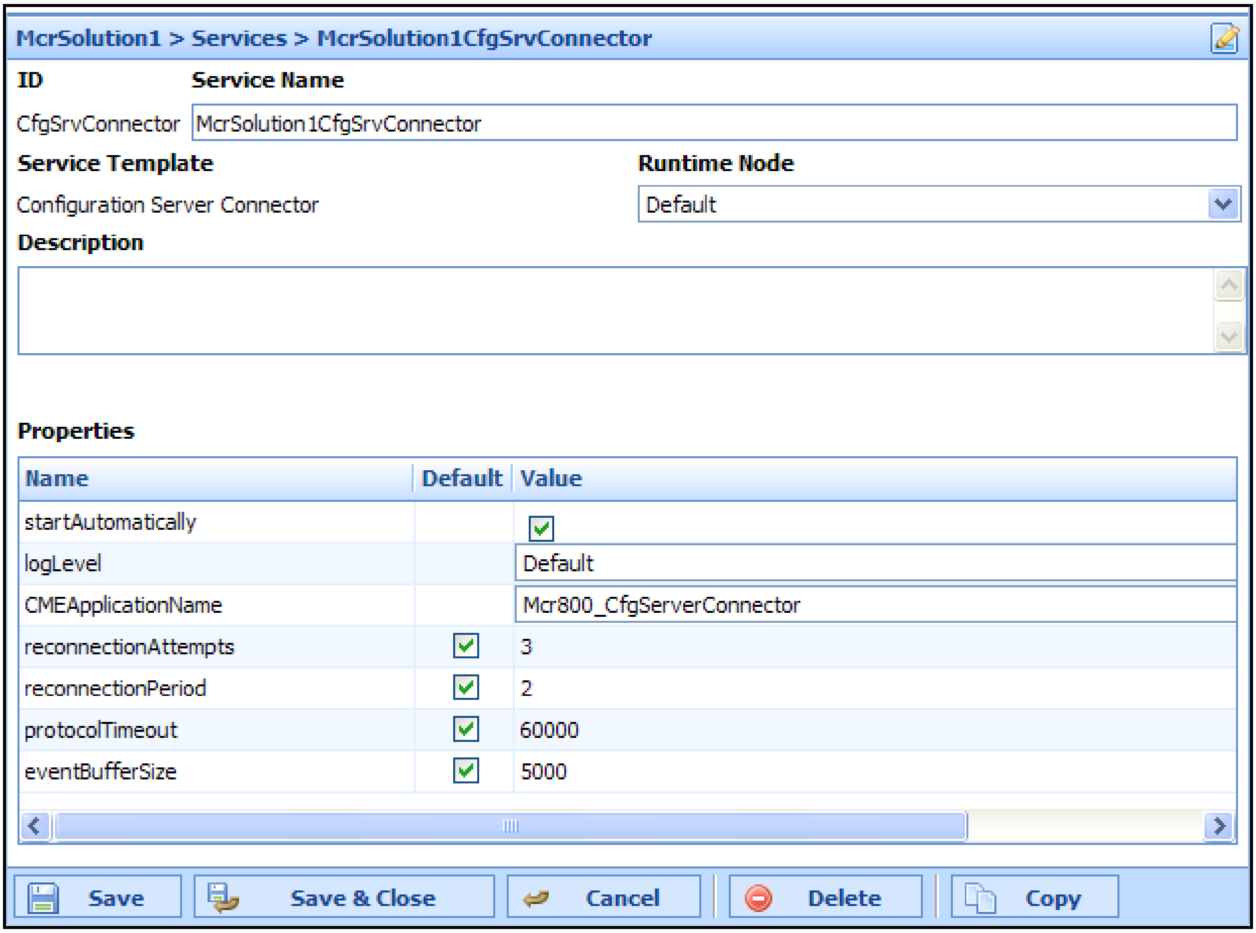
Attribute/Action	Description
Runtime Node	The runtime node on which the service is running. This field is absent for any service that runs on all available runtime nodes.
Service Name	The name of the service. Click here to open the Service Details view (see page 171).
▶	Starts a stopped service.
■	Stops a started service.

Table 17: Services View Properties and Actions (Continued)

Attribute/Action	Description
Status	Service status: • `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 177) for the specified service.

Service Details

Service Details view (see Figure 56) displays, and lets you modify, service attributes. See also “iWD Services” on page 178 for a detailed reference on specific service types.



McrSolution1 > Services > McrSolution1CfgSrvConnector

ID: CfgSrvConnector

Service Name: McrSolution1CfgSrvConnector

Service Template: Configuration Server Connector

Runtime Node: Default

Description:

Properties

Name	Default	Value
startAutomatically		☑
logLevel		Default
CMEApplicationName		Mcr800_CfgServerConnector
reconnectionAttempts	☑	3
reconnectionPeriod	☑	2
protocolTimeout	☑	60000
eventBufferSize	☑	5000

Figure 56: Service Details

Table 18 describes the properties and actions that are available in the Service Details view.

Table 18: Service Details View Properties and Actions

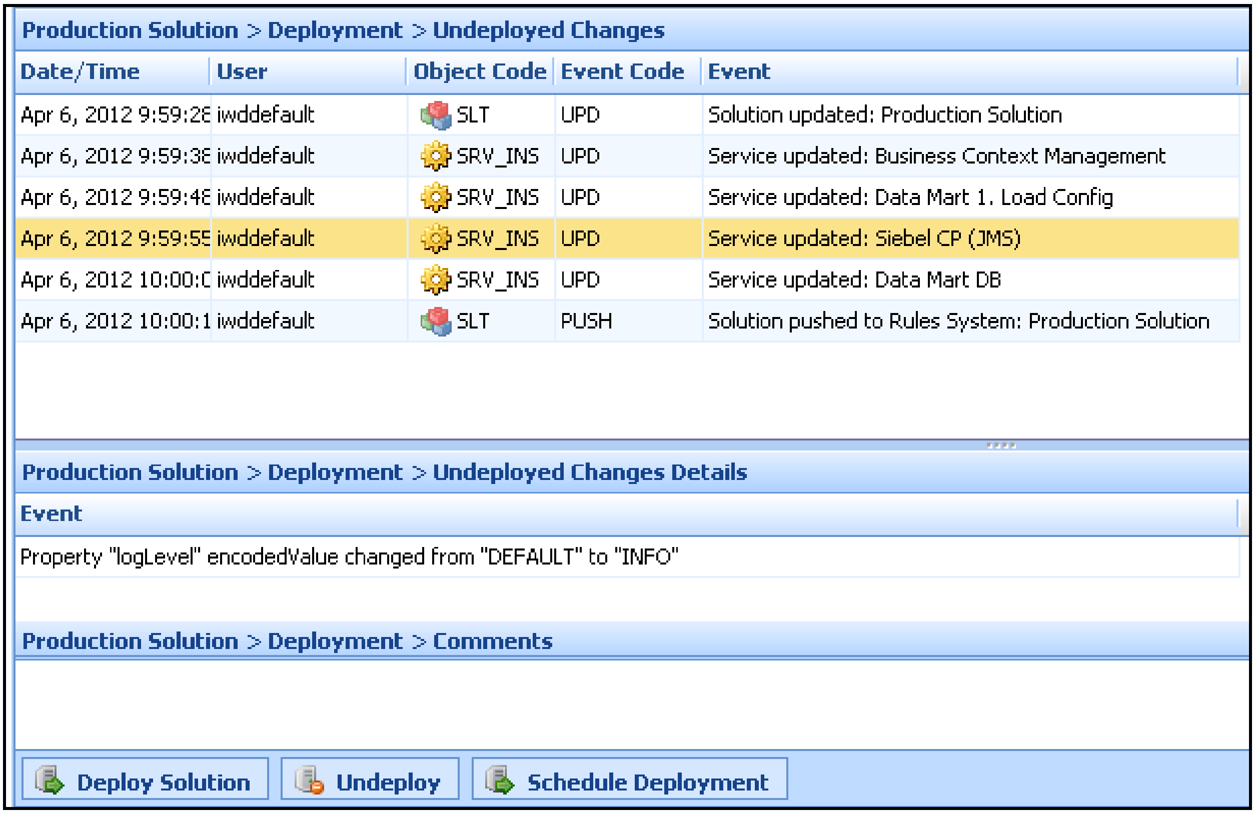
Property/Action	Description
ID	The ID of the service. This field becomes read-only after the service has been created. When a new service is created in iWD configuration, the default ID generated by iWD is based on the type of service being created. For example, if a new Scheduled ETL Job service is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

Table 18: 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 162 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	The properties of the service that configure its functionality: • `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. Different service types have different properties. For a detailed reference on specific service types, see “iWD Services” on page 178.
Save/Save & Close/Cancel/ Delete	Standard iWD Manager functions as described in “Object Actions” on page 138.
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. The Deploy functionality performs this task by distributing a solution's configuration across the defined runtime nodes. (see Figure 57 on page 174)



Production Solution > Deployment > Undeployed Changes

Date/Time	User	Object Code	Event Code	Event
Apr 6, 2012 9:59:28	iwddefault	SLT	UPD	Solution updated: Production Solution
Apr 6, 2012 9:59:38	iwddefault	SRV_INS	UPD	Service updated: Business Context Management
Apr 6, 2012 9:59:48	iwddefault	SRV_INS	UPD	Service updated: Data Mart 1. Load Config
Apr 6, 2012 9:59:59	iwddefault	SRV_INS	UPD	Service updated: Siebel CP (JMS)
Apr 6, 2012 10:00:0	iwddefault	SRV_INS	UPD	Service updated: Data Mart DB
Apr 6, 2012 10:00:1	iwddefault	SLT	PUSH	Solution pushed to Rules System: Production Solution

Production Solution > Deployment > Undeployed Changes Details

Event
Property "logLevel" encodedValue changed from "DEFAULT" to "INFO"

Production Solution > Deployment > Comments

Figure 57: Deploy

Note: If you see an on-screen notification, such as `There are changes to be pushed to Rules System`, you must push the changes to the Genesys Rules System before you deploy the rest of the iWD solution.

Table 19 describes the properties and actions that are available in the Deploy function.

Table 19: 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 160.
Comments	Deployment comments. They will be displayed as version comments in the Change History (see page 176).

Table 19: Deploy Properties and Actions (Continued)

Property/Action	Description
Deploy	Deploys the configuration. Depending on the configuration's complexity and runtime environment characteristics, this action can take several seconds. **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. All configured runtime nodes must be available to deploy. Deployment generates and copies a set of configuration XML 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.
Undeploy	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. All configured services are deactivated. Deployed configuration files are removed and the runtime node reloads the empty configuration.
Schedule Deployment	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. **Note:** If you have more than one iWD Manager instance connected to a single iWD Configuration database, there `iwd.host` parameter in the `iwd.properties` file must be unique for every host on which iWD Manager is running. The iwd.properties file can be found in the web application server's `webapps` folder for iWD Manager. For example, for Tomcat this might be `<TomcatDirectory>\webapps\iwd_manager\WEB-INF\classes`

Configure Interaction Custom Properties

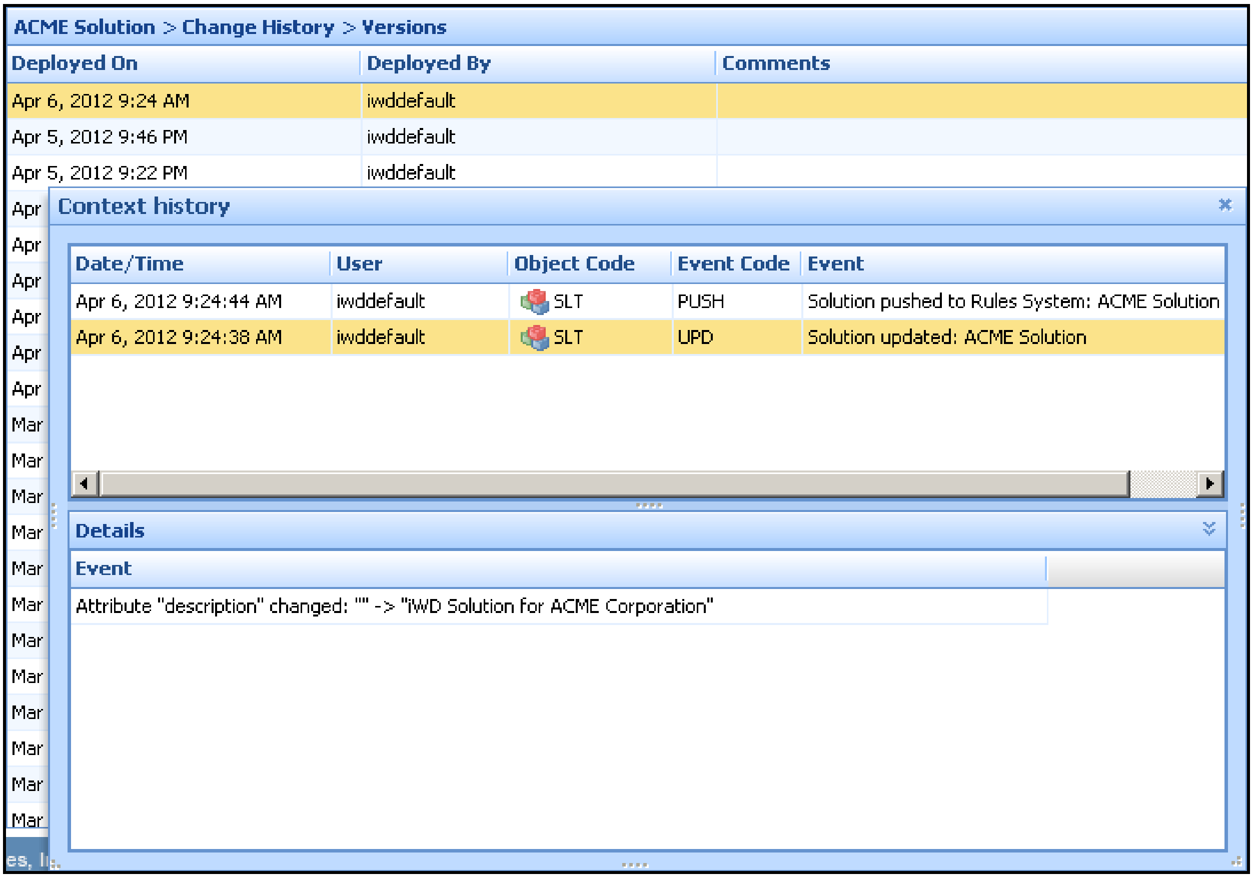
Configure Interaction Custom Properties supports iWD by adding the necessary columns to tables in the Interaction Server and Interaction Server

Event Log databases. It also creates new iWD-related Interaction Custom Properties, which are a type of Business Attribute, to Genesys Configuration Server. Finally, it creates some options for the Interaction Server Event Log Database Access Point that are necessary to support iWD.

If any of these updates were not previously done, clicking on the `Configure Ixn Custom Properties` node will display an on-screen message to the user, informing them of the missing attributes or outdated database versions. At this point the user must click on the `Configure Ixn Custom Properties` button, which will execute the configuration. A message will be displayed in the Messages pane when the configuration has completed. Interaction Server must be restarted if any configuration changes were made.

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 58 shows an example.



ACME Solution > Change History > Versions

Deployed On	Deployed By	Comments
Apr 6, 2012 9:24 AM	iwddefault	
Apr 5, 2012 9:46 PM	iwddefault	
Apr 5, 2012 9:22 PM	iwddefault	
Apr		
Apr		
Apr		
Apr		
Apr		
Apr		
Mar		
Mar		
Mar		
Mar		
Mar		
Mar		
Mar		
Mar		
Mar		
Mar		
Mar		
Mar		

Context history

Date/Time	User	Object Code	Event Code	Event
Apr 6, 2012 9:24:44 AM	iwddefault	SLT	PUSH	Solution pushed to Rules System: ACME Solution
Apr 6, 2012 9:24:38 AM	iwddefault	SLT	UPD	Solution updated: ACME Solution

Details

Event
Attribute "description" changed: "" -> "iWD Solution for ACME Corporation"

Figure 58: Change History

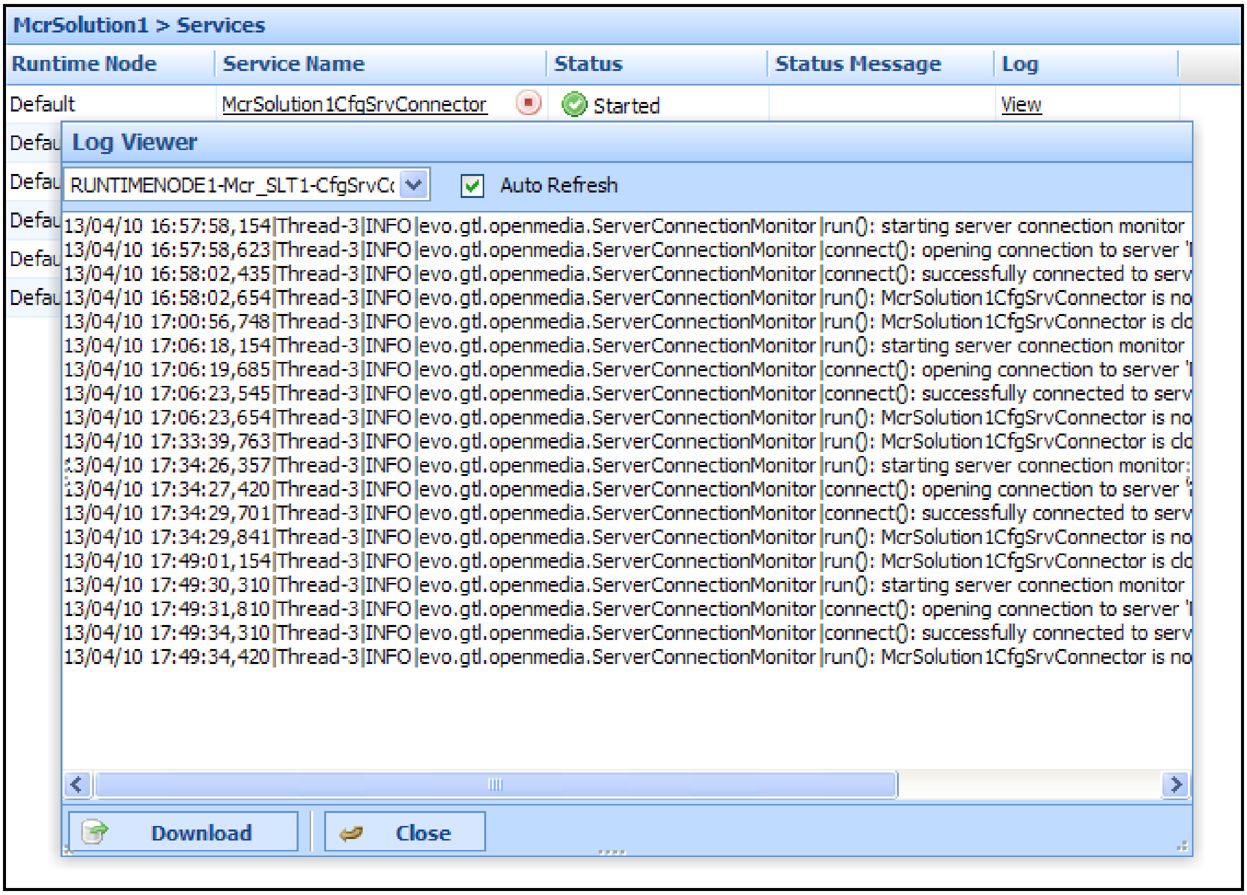
Table 20 describes the attributes and actions that are available in Change History view.

Table 20: 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 59) 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.



McrSolution1 > Services

Runtime Node	Service Name	Status	Status Message	Log
Default	McrSolution1CfgSrvConnector	Started		View
Defau				
Defau				
Defau				
Defau				
Defau				

Log Viewer

RUNTIMENODE1-Mcr_SLT1-CfgSrvC

☑ Auto Refresh

```
13/04/10 16:57:58,154|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |run(): starting server connection monitor
13/04/10 16:57:58,623|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |connect(): opening connection to server '
13/04/10 16:58:02,435|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |connect(): successfully connected to serv
13/04/10 16:58:02,654|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |run(): McrSolution1CfgSrvConnector is no
13/04/10 17:00:56,748|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |run(): McrSolution1CfgSrvConnector is clo
13/04/10 17:06:18,154|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |run(): starting server connection monitor
13/04/10 17:06:19,685|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |connect(): opening connection to server '
13/04/10 17:06:23,545|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |connect(): successfully connected to serv
13/04/10 17:06:23,654|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |run(): McrSolution1CfgSrvConnector is no
13/04/10 17:33:39,763|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |run(): McrSolution1CfgSrvConnector is clo
13/04/10 17:34:26,357|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |run(): starting server connection monitor
13/04/10 17:34:27,420|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |connect(): opening connection to server '
13/04/10 17:34:29,701|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |connect(): successfully connected to serv
13/04/10 17:34:29,841|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |run(): McrSolution1CfgSrvConnector is no
13/04/10 17:49:01,154|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |run(): McrSolution1CfgSrvConnector is clo
13/04/10 17:49:30,310|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |run(): starting server connection monitor
13/04/10 17:49:31,810|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |connect(): opening connection to server '
13/04/10 17:49:34,310|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |connect(): successfully connected to serv
13/04/10 17:49:34,420|Thread-3|INFO |evo.gtl.openmedia.ServerConnectionMonitor |run(): McrSolution1CfgSrvConnector is no
```

Figure 59: Log Viewer

iWD Services

iWD services implement actual iWD functionality, such as loading data into the Data Mart. iWD Services are classified as Core Services 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 4, "Installation," on page 53 for more information.

Core iWD Services

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

- 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—An ESP service called before and after the Genesys Rules Engine is invoked from a routing strategy, to ensure integrity of interaction user data.
- Generic Capture Point Service—Provides an iWD service that is used in conjunction with an Integrated Capture Point. An Integrated Capture Point is a service provided by Genesys Interaction Server that provides an easy way to submit interactions to Interaction Server (also known as, creating a task in iWD). Integrated Capture Points can also be used to manage submitted tasks, such as holding/resuming, canceling, completing, modifying, or restarting.

The Generic Capture Point service that is deployed within an iWD solution is used to provide an iWD Capture Point ID for the Integrated Capture Point to support task viewing and management through the Global Task List, and task reporting in iWD Data Mart.

For information about Integrated Capture Points see “Integrated Capture Points” on page 95.

Note: Genesys recommends that you use the Integrated Capture Points that are integrated into Interaction Server, instead of the capture point services style that is used with previous versions of iWD. However, the legacy capture point services are still available for use with iWD 8.1. For more information about legacy capture points, see Appendix B on page 277.

Reporting Services

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 21 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 8.1 Data Mart Reference Guide* for more information.

Table 21: 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.

Table 21: Preconfigured ETL Jobs (Continued)

Job Function	Attribute	Description
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.
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.

Table 21: Preconfigured ETL Jobs (Continued)

Job Function	Attribute	Description
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.
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 8.1 supports Genesys Workforce Management objects. A Workforce Management Connector service that uses the template `Workforce Management Connector` enables you to see some objects that exist in the WFM database, within business rules.

For more information about WFM objects and services, see "Workforce Management Connector Service" on page 200.

Recommended Order of Configuration for iWD Services

Table 22 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 22: iWD Services

Name	Mandatory	Category	Dependencies	Notes
Logging Service	Yes	Core	None	See page 183.
Configuration Server Connector	Yes	Core	None	See page 184.

Table 22: iWD Services (Continued)

Name	Mandatory	Category	Dependencies	Notes
Interaction Server Connector	Yes	Core	Genesys Configuration Server Connector	See page 185.
Business Context Management Service	Yes	Core	Genesys Configuration Server Connector	See page 187.
Generic Capture Point Service	Yes	Core	Interaction Server Connector	See page 185
Database Service	No	Reporting	None	See page 188.
Statistics Adapter	No	Reporting	None	See page 190.
Kettle ETL Service	No	Reporting	Database	See page 193.
Scheduled ETL Job	No	Reporting	Kettle ETL Service	A separate scheduled ETL job that must be created for each job type. See page 196.
WFM Connector	No	Workforce Management	Genesys Configuration Server Connector	See page 200.

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 183).
- `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 171, the properties listed in Table 23 are configurable for the Logging Service.

Table 23: Logging 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 `Info` unless otherwise instructed by Genesys Technical Support. See “Service Log Levels” on page 182 for a description of each log level.
logAgeInDays	Sets the number of days that log files should be kept in the system. A value of `0` disables this limit. This property is not available if `logLevel` is set to `Off`.
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	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. **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.

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 171, the properties listed in Table 24 are configurable for the Configuration Server Connector service.

Table 24: 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 182 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 183 for descriptions of these common properties.
reconnectionAttempts	Exception handling: The maximum number of attempts to connect to Configuration Server, before connecting to the backup server.
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 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 171, the properties listed in Table 25 are configurable for the Interaction Server Connector service.

Table 25: 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 182 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 183 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.

Table 25: Interaction Server Connector Service Properties (Continued)

Property	Description
attributeFilerInclude	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"
attributeFilterExclude	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. 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" **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.
configurationServerConnector	Mandatory dependency: The Configuration Server Connector service (see page 187); provides access to the Configuration Server.

Table 25: Interaction Server Connector Service Properties (Continued)

Property	Description
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 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.

Business Context Management Service

The Business Context Management service is a mandatory service that is called from a routing strategy before and after the Genesys Rules Engine is invoked. The Business Context Management service is invoked to prepare the interaction user data prior to the Genesys Rules Engine being called, or after the Genesys Rules Engine has been called, depending on the ESP method that has been requested.

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 266 for information about configuring multiple Business Context Management Services.

In addition to the options and actions described in “Service Details” on page 171, the properties listed in Table 26 are configurable for this service.

Table 26: 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 182 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 183 for descriptions of these common properties.

Table 26: Business Context Management Service Properties (Continued)

Property	Description
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 105.
configurationServerConnector	Mandatory dependency: The Configuration Server Connector service (see page 187); provides access to the Configuration Server.
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.

Generic Capture Point Service

The Generic Capture Point service provides a way to instantiate an iWD Capture Point ID for an Interaction Server Integrated Capture Point.

In addition to the options and actions that are described in “Service Details” on page 171, the properties listed in Table 27 are configurable for the Generic Capture Point Service.

Table 27: Generic Capture Point Service Properties

Property	Description
startAutomatically	Indicates whether or not the service is started automatically after the configuration deployment.
logLevel	The Service log level. Set this property to `Default` unless otherwise instructed by Genesys Technical Support. For a description of each log level, see “Service Log Levels” on page 182. Depending on the setting of this property, additional logging properties might be available. For descriptions of these common properties, see to “Logging Service” on page 183.
interactionServerConnector	Mandatory dependency. The Interaction Server Connector service (see page 185) that must be used for connectivity to the Interaction Server.

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 171, the properties listed in Table 28 are configurable for the Database Service.

Table 28: 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 `Default` unless otherwise instructed by Genesys Technical Support. See “Service Log Levels” on page 182 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 183 for descriptions of these common properties.
DatabaseType	The type of the database server. Refer to the *Genesys Supported Operating Environment Reference Manual* 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 `MySQL`.
SqlServerCustomURL	If selected, the connection is configured by specifying a custom MS SQ JDBC URL. This option is available only when the `DatabaseType` is `MS SQL Server`.
SqlServerJDBCURL	The JDBC connection URL for MS SQL. This option is available when `SqlServerCustomURL` is selected.
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`.

Table 28: Database Service Properties (Continued)

Property	Description
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 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.

The Statistics Adapter service will create the following statistics in the Stat Server configuration by default:

- The number of Active tasks for each Department and Process
- The number Held tasks for each Department and Process
- The number of New tasks, as of the trailing 15, 30, and 60 minute intervals, for each Department and for the Solution
- The number of Completed tasks, as of the trailing 15, 30, and 60 minute intervals, for each Department and for the Solution
- The number of Pending tasks, as of the trailing 15-minute interval, for each Department and Process
- The number of Overdue tasks, as of the trailing 15-minute interval, for each Department and Process

In addition to the options and actions described in "Service Details" on page 171, the properties listed in Table 29 are configurable for the Statistics Adapter Service.

Table 29: 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 182 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 183 for descriptions of these common properties.
triggerMode	How the ETL job will be triggered (started): • `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 55 on page 170). **triggerMode and startAutomatically** 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. 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. **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. If `startAutomatically` is checked and the `triggerMode` is set to `Trigger using CRON expression`, the service will start according to the CRON expression.

Table 29: Statistics Adapter Service Properties (Continued)

Property	Description
cronExpression	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: `0 0,15,30,45 * * * ?` For more information about CRON scheduling, see `http://www.quartz-scheduler.org/documentation/quartz-2.1.x/tutorials/crontrigger`.
executionQueueName	The execution queue is configured only for jobs that are triggered with a cron expression or triggered manually. 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. Scheduled services that have the same `executionQueueName` will never be run in parallel within the same solution. 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. **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.
dimensionMapping	Defines how statistical dimensions are mapped. `Filter`: Dimensions are mapped to CCPulse+ filters. `Virtual` Queue: Dimensions are mapped to Genesys virtual queues.
datamartDatabase	A reference to the Database service (see page 188) 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 184); 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 29: 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 `java-libraries-dir` option in the `java-config` 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 171, the properties listed in Table 30 are configurable for the Kettle ETL service.

Table 30: 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 182 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 183 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	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. 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: `Name: <attribute_name>, type: varchar(length).` These fields should be added to the mappings in the Event Log DAP options in the `esp-custom-data` and `itx-custom-data` sections. For example, in order to store a custom attribute with the name `order_total` in the iWD Data Mart, as a task custom attribute: 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 30: Kettle ETL Service Properties (Continued)

Property	Description
customTaskDimensionMapping	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. 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: Name: <attribute_name>, type: varchar(length). These fields should be added to the mappings in the Event Log DAP options in the esp-custom-data and itx-custom-data sections. For example, in order to store a custom attribute with the name order_total in the iWD Data Mart, as a task custom attribute: 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 30: Kettle ETL Service Properties (Continued)

Property	Description
datamartDatabase	Mandatory dependency: Database (see page 188) in which to load and aggregate reportable data (Data Mart).
statisticsAdapter	Optional dependency: Statistics Adapter service (see page 190) 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 171, the properties listed in Table 31 are configurable for the Scheduled ETL Job service.

Table 31: 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 182 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 183 for descriptions of these common properties.

Table 31: Scheduled ETL Job Service Properties (Continued)

Properties	Description
triggerMode	How the ETL job will be triggered (started): • `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 55 on page 170). **triggerMode and startAutomatically** 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. 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. **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. If `startAutomatically` is checked and the `triggerMode` is set to `Trigger using CRON expression`, the service will start according to the CRON expression.
cronExpression	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: `0 0,15,30,45 * * * ?` For more information about CRON scheduling, see `http://www.quartz-scheduler.org/documentation/quartz-2.1.x/tutorials/crontrigger`.

Table 31: Scheduled ETL Job Service Properties (Continued)

Properties	Description
executionQueueName	The execution queue is configured only for jobs that are triggered with a cron expression or triggered manually. 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. Scheduled services that have the same `executionQueueName` will never be run in parallel within the same solution. 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. **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.
etlService	Mandatory dependency: The Kettle ETL Service (see page 193)
jobName	The name (type) of the ETL job. See Table 21, “Preconfigured ETL Jobs,” on page 179 for descriptions of the various types of iWD Data Mart ETL jobs. Additional information can be found in the *iWD 8.1 Data Mart Reference Guide*.

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 199

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 105.

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 Genesys 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 8.1 Administrator's Guide.*
- The `attributeFilterExclude` property of the Interaction Server Connector service should contain the `WFM Activity` parameter (see Table 25 on page 185 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 171 for the common service options. Table 32 lists the additional configurable properties for the Workforce Management Connector Service.
6. Once the service is configured, WFM objects can be used in iWD business rules.
7. In the Genesys Rules Authoring Tool, 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 32: 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 182 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 183 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 187); 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.

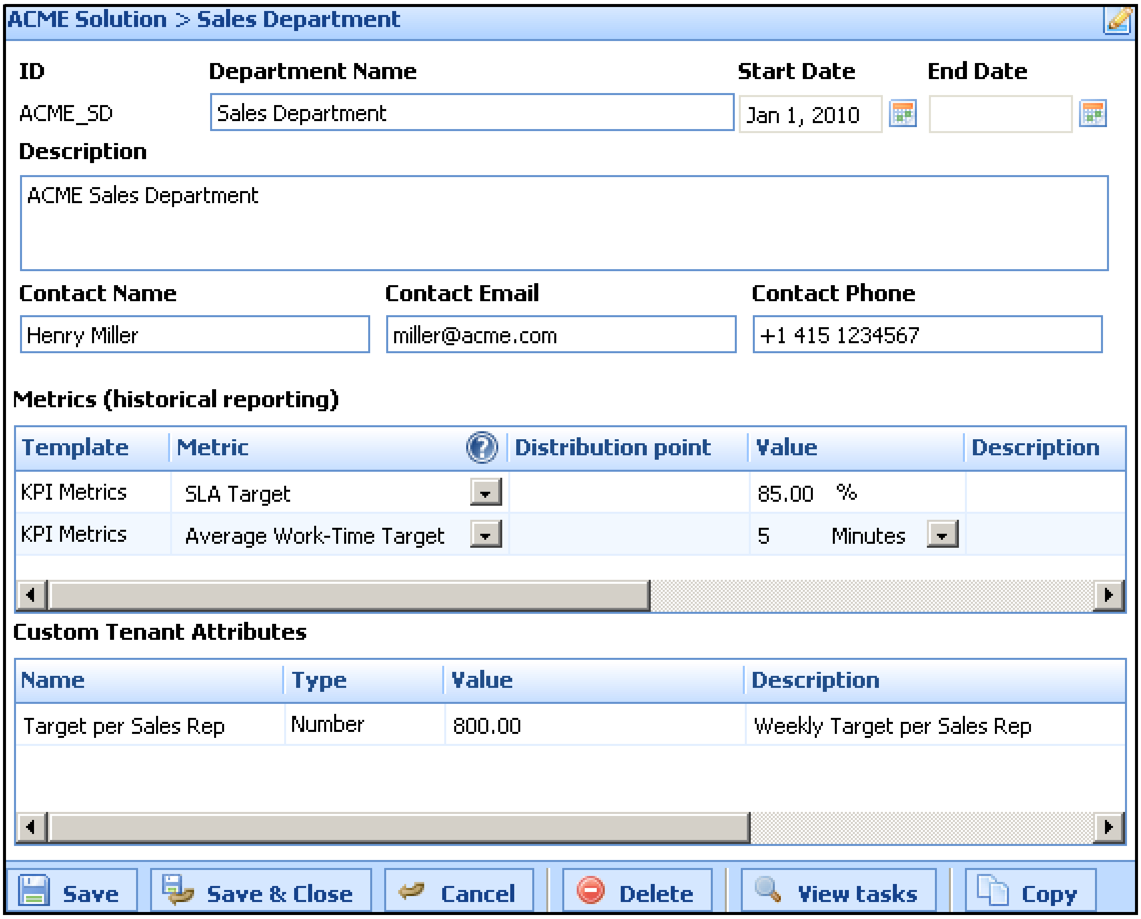
- Use Department Details view (see page 203) to create new departments, and to view and modify general department attributes and contract-level Metrics (see page 206).
- Use Process Details view (see page 204) to create new processes, and to view and modify general process attributes and process-level Metrics (see page 206).
- Use Deployment view (see page 207) to deploy solution configurations.
- Use User Access view (see page 207) to control which security roles can access specific Departments and Processes.

- Use Push to Rules System (see page 207) to push iWD business structure changes to Genesys Configuration Server, to synchronize them for use in the Genesys Rules System.

Note: Starting with iWD 8.1, Business Calendars and Rules are no longer configured in iWD Manager under Departments & Processes. Business Calendars and all business rules are created, modified, and deployed through the Genesys Rules Authoring Tool. For more information about Genesys Rules System see Chapter 6 on page 209.

Department Details

Department Details view displays, and lets you modify, general department attributes. Refer to “Departments and Processes” on page 202 for more information about departments. Figure 60 shows the Department Details view



ACME Solution > Sales Department

ID	Department Name	Start Date	End Date
ACME_SD	Sales Department	Jan 1, 2010	

Description

ACME Sales Department

Contact Name	Contact Email	Contact Phone
Henry Miller	miller@acme.com	+1 415 1234567

Metrics (historical reporting)

Template	Metric	Distribution point	Value	Description
KPI Metrics	SLA Target		85.00 %	
KPI Metrics	Average Work-Time Target		5 Minutes	

Custom Tenant Attributes

Name	Type	Value	Description
Target per Sales Rep	Number	800.00	Weekly Target per Sales Rep

Figure 60: Department Details

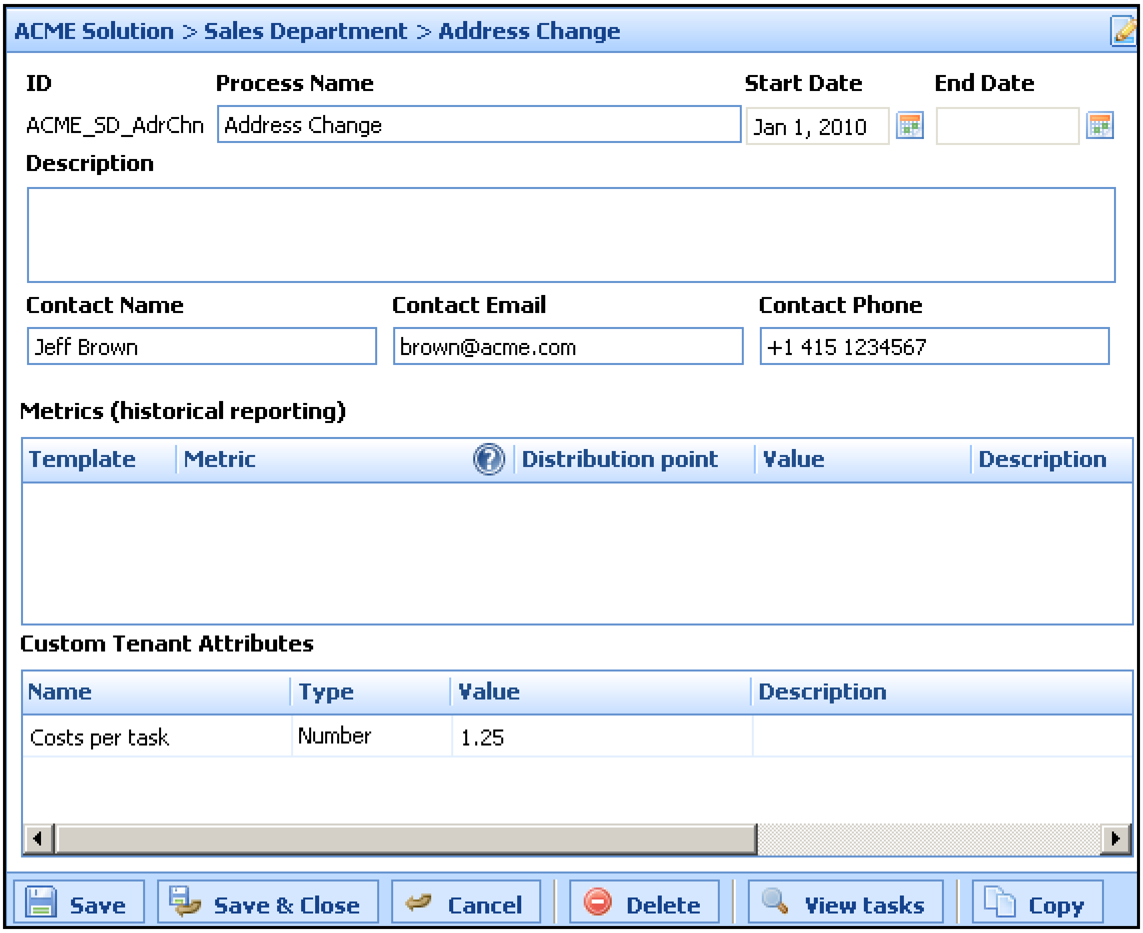
Table 33 lists the attributes and actions that are available in the Department Details view.

Table 33: 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 206).
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 138.
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 202 for more information about processes. Figure 61 shows the Process Details view.



ACME Solution > Sales Department > Address Change

ID	Process Name	Start Date	End Date
ACME_SD_AdrChn	Address Change	Jan 1, 2010	

Description

Contact Name	Contact Email	Contact Phone
Jeff Brown	brown@acme.com	+1 415 1234567

Metrics (historical reporting)

Template	Metric	Distribution point	Value	Description

Custom Tenant Attributes

Name	Type	Value	Description
Costs per task	Number	1.25	

Figure 61: Process Details

Table 34 lists the attributes and actions that are available in Process Details view.

Table 34: 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.
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.

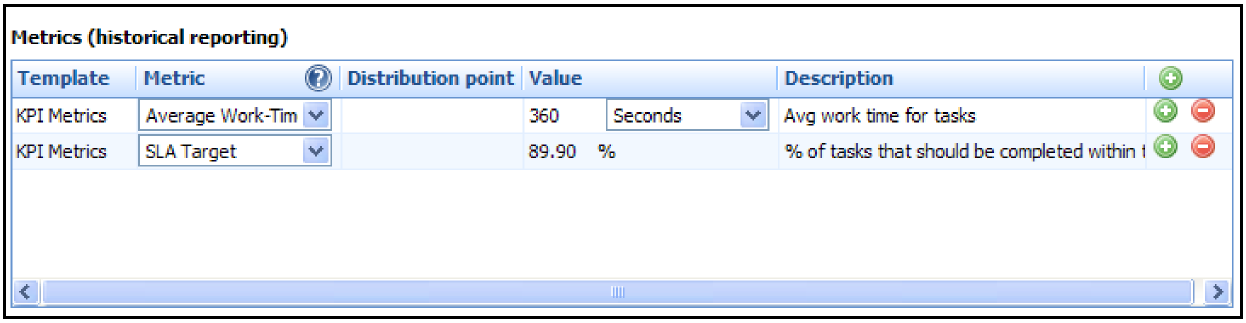
Table 34: Process Attributes and Actions (Continued)

Attribute/Action	Description
Metrics	A set of user-defined metrics, for reporting purposes (described in “Metrics” on page 206).
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 138.
View Tasks	Opens a corresponding task view in iWD Manager.

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 62 shows an example.



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

Figure 62: 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 “Metric Templates” on page 164). 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.

Deployment

Deployment in the Departments and Processes configuration section is identical to Deployment in the Services configuration section. Refer to "Deployment" on page 207 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.

Assigning Security Roles

To assign security roles to specific departments or processes, double-click the space to the right of the department or process name. A pop-up window will appear with a list of all configured security roles. To add a security role to the list, hold down the `Control` key and click the left mouse button. You can repeat that action to multi-select. You can remove security roles from the list in the same way.

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.

Push To Rules System

When a user has made changes to the iWD business structure, a notification appears on the top of the iWD Manager screen. The user can click the hyperlink on that notification to go to the `Push to Rules System` screen.

This screen has a single button named `Execute.` The user can click this button to push the iWD business structure (`Solution->Departments->Processes`) to Genesys Configuration Server, so that it can be available to the Genesys Rules System. This business structure synchronization is executed for the current iWD solution only.

Chapter

6 iWD and the Genesys Rules System

This chapter provides information about how iWD interoperates with the Genesys Rules System. It contains the following sections:

Introduction To Genesys Rules System

The Genesys Rules System (GRS) is a set of components that provides business rules functionality for use with the iWD solution as well as other Genesys solutions. It consists of three software components:

- Genesys Rules Development Tool
- Genesys Rules Authoring Tool
- Genesys Rules Engine

The general workflow for using Genesys Rules System is shown in Figure 63 on page 210.

Note: For information about how to migrate your existing rule templates and rules, see the *Genesys System Migration Guide.*

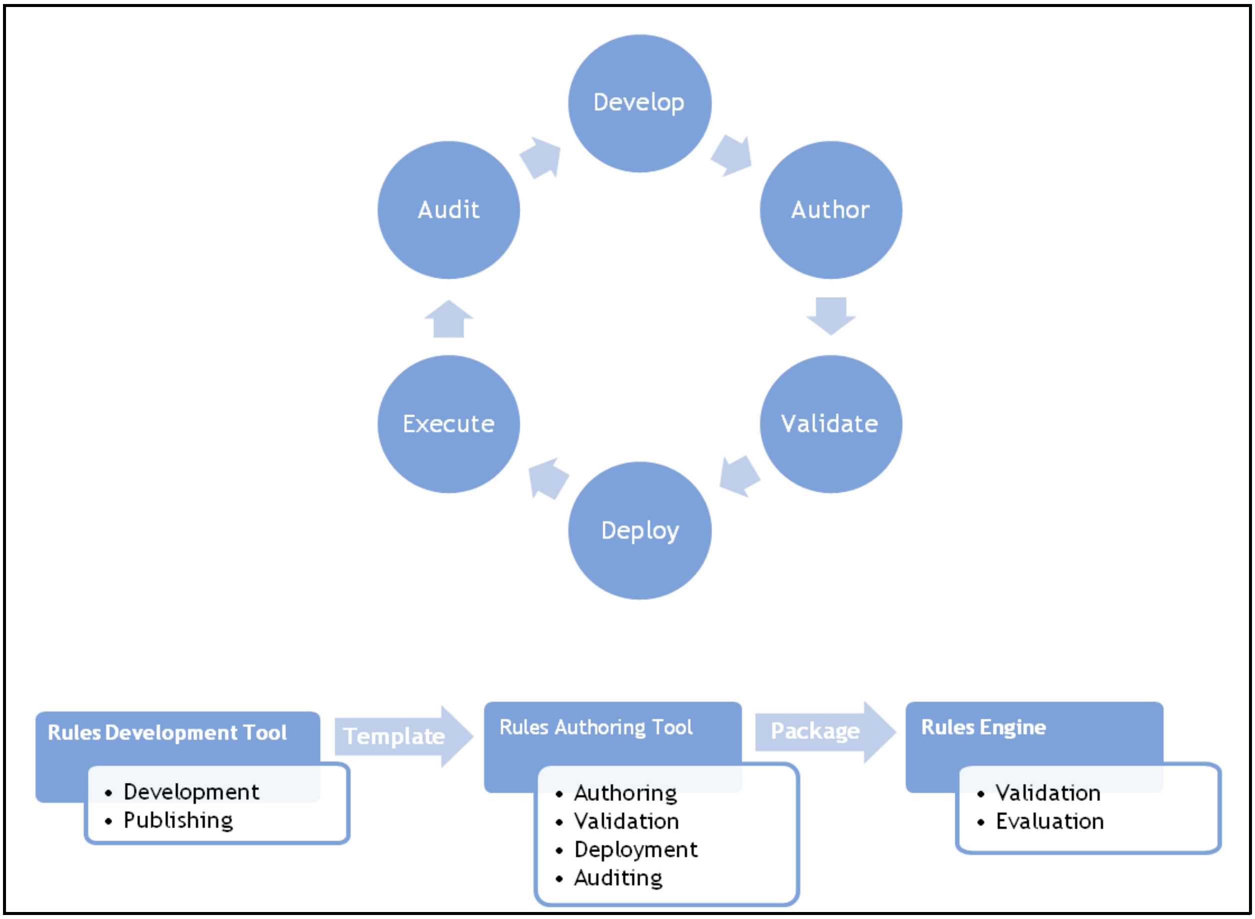


Figure 63: Genesys Rules System Lifecycle

Rule templates are created in the Genesys Rules Development Tool and the templates are published to the rules repository. Users then use the Genesys Rules Authoring Tool to create a rule package that incorporates one or more rule templates. The Rules Authoring Tool is also where users create new rule packages that incorporate rule templates, author rules inside the rule package based on the rule templates, validate the rules, and then deploy their rule package to the Genesys Rules Engine. Client applications such as the iWD business process (IWDBP) then make requests to the Genesys Rules Engine to have rules in the rule package evaluated at various decision points in a task's lifecycle.

The Genesys Rules System is not only used by Genesys iWD. It is also used by other Genesys solutions, including Genesys Conversation Manager. As such, there are some objects that must be properly configured for iWD, when working with the Genesys Rules System. These include configuring the proper rule template type when creating a new template, configuring the proper rule package type when creating a rule package, and the proper use of rule phases. These objects are described in more detail in this chapter.

Rule Templates

Rule templates for iWD are created in the Genesys Rules Development Tool, which is an Eclipse plug-in that can either be installed into a standalone Eclipse application or can be installed into Genesys Composer.

Rule templates are used to define the building blocks that are used by rules authors to build rules for task classification and prioritization at the Global, Department, and Processes levels of the business structure of an iWD Solution.

Rule templates are made up of four elements:

1. Conditions—Contain `when` or `if` expressions
2. Actions—Contain `then` expressions
3. Parameters—Used in the actions and conditions
4. Functions—Sometimes used to support rule conditions and actions. For example, when parsing of timestamps is required

Rule templates also include a fact model with one or more facts. All iWD rule templates include a fact called `_GRS_Environment`, which must not be modified. Unlike rule templates for other Genesys solutions, it is not necessary to create any additional facts for iWD rules.

For more information about the components of a rule template, see Chapter 3 in the *Genesys Rules System 8.1 Deployment Guide*.

Note: Rule conditions and actions are made up of a Language Expression and a Rule Language Mapping. The Language Expression defines what the rule condition or action does, in plain language. It is what the rules author (who is often a business user) will see when constructing a rule. The Rule Language Mapping defines the same Language Expression in Java code. This is the real code that is executed. Rule Language Mapping follows Drools 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. For more information, see the *Platform SDK Developer's Guide* (release 7.6 or later).

Normally rule templates are created or modified by IT personnel. Defining new rule conditions or actions sometimes requires a basic knowledge of the Java programming language, although in many cases it is possible to review the syntax of existing rule conditions and actions as a guide. After rule templates are added to a rule package, business users can create or modify rules by using the Genesys Rules Authoring Tool without having to involve IT personnel.

iWD comes with an out-of-the-box rule template called the iWD Standard Rules Template, that a user will import into the Genesys Rules Development Tool. This template is the foundation for the rules that are most frequently used in iWD deployments. The template can be used as an example when new rule conditions or actions are required for a particular iWD deployment.

Although the functionality defined in the iWD Standard Rules Template can be extended, Genesys recommends creating one or more custom rule templates for this purpose. One reason for this is that you can use Configuration Server object permissions to control which users can access specific rule templates. For example, you might have one custom rule template for all users of the Genesys Rules Authoring Tool, which includes common rule conditions and actions, to which all users should have access. Then you can have a second custom rule template that contains advanced rule conditions and actions, which you only make accessible to experienced rule authors.

Note: If Genesys provides an updated iWD Standard Rules Template, all custom changes to the previous version will be lost after importing of the updated template.

After you modify an existing rule template, or create a new one, the rule template is published to the Genesys Rules System rules repository. This makes the template available to users of the Genesys Rules Authoring Tool, so the template can be added to a new rule package, or an existing rule package can be updated with the latest version of a template.

The following procedures describe how to import the iWD Standard Rules Template into the Genesys Rules Development Tool, and how to create a custom rule template for your iWD solution, if necessary.

Procedure:
Importing the iWD Standard Rules Template into the Genesys Rules Development tool and publishing it to the Rules repository

Purpose: To import the iWD Standard Rules Template into the Genesys Rules Development Tool and then publish it to the Genesys Rules System rules repository. This will make the Standard Rules Template available to rules authors, from within the Genesys Rules Authoring Tool.

Prerequisites

- The Genesys Rules System components are installed and configured according to the procedures in Chapter 2 in the *Genesys Rules System 8.1 Deployment Guide.*

Start of procedure

1. Launch Eclipse (or Composer).
2. Open the Template Development perspective (if it is not already open).
3. Navigate to `Window > Open Perspective > Other > Template Development`.
4. In the `Eclipse` menu, navigate to `File >Import`.
5. In the `Import` dialog box, select Existing Projects into Workspace from within the `General` folder.
6. Click `Next`.
7. Select the `Select root directory` radio button
8. Select `Browse` to browse to the iWD Standard Rules Template folder.

Note: The iWD Standard Rules Template will be in the rule`Templates` subdirectory in the folder where the iWD Manager supporting files were installed on your file system. For example, `C:\Program Files\GCTI\iWD Manager\ruleTemplates`.

9. Click `Finish`.

   The iWD Standard Rules Template is now visible in the Eclipse (or Composer) `Project Explorer` view.
10. In the `Project Explorer` view, right-click `iWD_Standard_Rules project` and select `Publish`·
11. Using the `Publish Template Wizard`, select the `iWD_Standard_Rules` template project.
12. Select the `Edit properties` link.
13. In the `Properties` dialog, under `Template Properties`, ensure that `Type = iWD`.
14. For the `Tenant` property, select the Configuration Server tenant to which this rule template will be associated.
15. Click `OK`.
16. Using the `Publish Template Wizard`, select the `iWD_Standard_Rules` template project.

17. Click `Finish`.

    The `Publishing` dialog, with a progress indicator, will display `Publish successful` after the template project has been successfully published to the Genesys Rules System repository.

Note: You cannot import the iWD Standard Rules Template project into GRDT if it already exists in the Eclipse workspace. Therefore, if you have a new version of the iWD Standard Rules Template to import, you must first delete the existing one. For this reason, Genesys strongly recommends that you use custom rule templates for new rule conditions or actions, rather than extending the iWD Standard Rules Template.

End of procedure

Next Steps

- If you do not plan to create any custom rule templates at this time, go to the Genesys Rules Authoring Tool to create a rule package. See "Rule Authoring" on page 229.
- To review information about the parameters, conditions, actions, and functions provided in the iWD Standard Rules Template, see "About the iWD Standard Rules Template" on page 217.

Procedure: Creating a custom rule template to use with iWD

Purpose: To create a custom rules template to use with iWD.

Prerequisites

- The Genesys Rules System components are installed and configured according to the procedures in Chapter 2 in the *Genesys Rules System 8.1 Deployment Guide.*

Start of procedure

1. Start the Genesys Rules Development Tool.
2. To open the `Template Development` perspective (if not already opened); go to `Window > Open Perspective > Template Development`.

   Note: If the `Template Development` perspective does not appear in the list, select `Other` and then, select the `Template Development` perspective.

3. Select `File > New > Rule Template Project`.

Note: If the `Rule Template Project` type is not visible, select `Other`, and then, within the Genesys Rules System folder in the `New` dialog, select `Rule Template Project`.

4. Click `Next`.
5. On the `Rule Template Project` screen of the `New Rule Template Project` dialog, enter a name for the template project and select the Eclipse workspace where the project files will be stored.
6. Click `Next`.
7. On the `Template Properties` screen, for the `Type` property, select `iWD`.
8. For the `Tenant` property, select the Configuration Server tenant to which this rule template will be associated.

Note: Optionally, you can provide a description for the rule template project.

9. Click `Finish`.

   The custom rule template is displayed in the `Eclipse Project Explorer` view.

Next Steps

- After parameters, conditions, actions, and functions are created as necessary for the custom rule template, you must publish it to the Genesys Rules System rules repository. See steps 8-15 of the Procedure: Importing the iWD Standard Rules Template into the Genesys Rules Development tool and publishing it to the Rules repository, on page 212.

For more information about rule template parameters, conditions, actions, and functions, see the *Genesys Rules System 8.1 Deployment Guide* and the *Genesys Rules System 8.1 Development Tool Help.*

Custom Rule Template Conditions and Actions

The tips provided in this section can be useful when you are creating new rule parameters, conditions, and actions to use with iWD.

Language Expression Syntax

If a rule action Language Expression contains one or more parameters, the expression must terminate with one of those parameters. For example, the following Language Expression for a rule action generates an error:

```
Set Priority "{priority}" and update
```

Whereas, the following Language Expression is valid:

```
Set and Update Priority "{priority}"
```

Using iWD Lookup Tables in Rules

The Genesys Rules System offers several ways to create rule parameters that will present an enumerated list of items to the rules author as a drop-down list. However, there are certain situations, in which you might want to use a Lookup Table that has been configured in iWD Manager, as the basis for a dynamic rule parameter. This might be valuable because Lookup Tables provide user-friendly name/label pairs that can be changed easily by a business user, while some other types of dynamic parameters, such as List Objects in Configuration Server, cannot be edited easily by a business user.

If you want to use a Lookup Table in a rule parameter, it will be a parameter of type `database`, where the database being read is the iWD configuration database. The best way to create a new rule parameter that will look up values from an iWD Manager Lookup Table is to use an existing parameter as an example. For example, in the iWD Standard Rules Template, you can look at the `taskChannels` rule parameter and, in the ACME Rules Template, you can look at the `ACMEproduct` parameter.

Using the `ACMEproduct` parameter as an example, we can see that the SQL query that is executed will read data from several tables in the iWD configuration database:

```
SELECT cmc_lookup_attr_entry.valueKey,cmc_lookup_attr_entry.valueLabel
FROM  ((cmc_lookup_attr_entry INNER JOIN cmc_lookup_attr_type ON
cmc_lookup_attr_entry.attributeTypeId=cmc_lookup_attr_type.id) INNER
JOIN cmc_tenant ON cmc_lookup_attr_type.tenantId=cmc_tenant.id) WHERE
cmc_lookup_attr_type.name='Products' AND cmc_tenant.name='ACME';
```

In this example, the only parts of the query that would need to change are the name of the Lookup Table, which is referenced in the query as `cmc_lookup_attr_type.name`, and the name of the tenant, which is referenced in the query as `cmc_tenant.name`. If the Lookup Table is configured under the System tenant, then the `cmc_tenant.name` would be set to `System` in the query.

Operational Parameters

Operational parameters provide another user-friendly way to make rules dynamic, enabling a business user to change a dynamic value that will be referenced in a rule, without having to change the rule itself. Operational parameters are a special type of parameter that is created by users through Genesys Administrator Extension and, when deployed, are stored as options of `Transaction` objects of the type `List` in the Genesys Configuration Server database. At rule execution time, when the Genesys Rules Engine evaluates a rule that contains an operational parameter, it obtains the current value of the associated `Transaction` object option from Configuration Server. The rule developer determines from which `Transaction` object, and which option of that object, the value of the operational parameter should be fetched, and the rule author uses this parameter as part of a condition or action.

For example, an operational parameter might be called `waitTimeThreshold`. If a caller is waiting longer than this threshold for an available agent, perhaps some other action will be performed.

Instead of specifying a value for the threshold in the rule like the following:

```
When
    Caller's wait time is greater than 30 seconds
Then
    Offer a callback
```

the rule author could specify:

```
When
    Caller's wait time is greater than {waitTimeThreshold}
Then
    Offer a callback
```

The value of `{waitTimeThreshold}` can be changed at any time by a user that is using Genesys Administrator Extension and this action will have an immediate effect without the user having to modify and redeploy a rule package.

About the iWD Standard Rules Template

The iWD Standard Rules Template defines the most commonly used rule conditions, actions, parameters, and functions.

Note: The ACME sample application (see “Sample Application” on page 126) contains sample 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 values based, on the iWD process. The sample also includes examples of how to set the reprioritization interval in a rule.

Conditions

Table 35 describes conditions for the Standard Rules Template.

Table 35: iWD Standard Rules Template Conditions

Condition	Explanation
Business value is "businessValue_From" to "businessValue_To"	If the business value of the task is between businessValue_From and businessValue_To, then... This rule condition is designed to test inclusive conditions. That is, the "businessValue_From" and "businessValue_To" conditions that are being checked are ">=" and "<=", respectively. This must be understood in order to use this rule condition effectively.
Capture point is "capturePoint"	If the capture point is capturePoint, then… This rule condition is designed to be used at the rule package (Global) level, as a way to classify tasks, based on the capture point from which they originated. This is in lieu of the Capture Point-level rules that were available in iWD 8.0 but are not available in the Genesys Rules System. This rule condition requires that the capturePoint rule parameter be properly configured so that GRAT can access the list of Capture Points from the iWD configuration database. See “Enabling Genesys Rules Authoring Tool to Retrieve Dynamic Data from iWD Configuration Database” on page 83.
Channel is "taskChannels"	If the specified channel of the task is taskChannels, then... This rule condition requires that the taskChannels rule parameter be properly configured so that GRAT can access the list of task channels from the iWD confutation database. See “Enabling Genesys Rules Authoring Tool to Retrieve Dynamic Data from iWD Configuration Database” on page 83.
Department is "department"	If the specified department of the task is department, then...
Due Time is in "periodFrom" to "periodTo" "periodType"	If the task due date/time is between periodFrom and periodTo specified time periodType, then... This rule condition is designed to test conditions that are not inclusive. That is, the "periodFrom" and "periodTo" conditions that are being checked are ">" and "<", respectively. This must be understood in order to use this rule condition effectively.
Integer "attribute" "operator" "integerValue"	When the value of a specified custom attribute of type integer is greater than/less than/equal to the specified integerValue, then...
Is first prioritization	When the rules being applied are part of the initial prioritization step (the value of the task attribute reprioritizeDateTime is empty), then…

Table 35: iWD Standard Rules Template Conditions (Continued)

Condition	Explanation
Is reprioritization	When the rules being applied are part of the reprioritization step (the value of the task attribute `reprioritizeDateTime` is not empty), then…
Is working day	When today is a working day, then… This condition is evaluated, based on the default business calendar assigned to the rule package or a business calendar that has been explicitly assigned to the task in a previous rule. A business calendar must be assigned to the rule or in a separate rule action before this action can be used.
Is working time	When the current time is working time, then… This condition is evaluated based on the default business calendar assigned to the rule package or a business calendar that has been explicitly assigned to the task in a previous rule. A business calendar must be assigned to the rule or in a separate rule action before this action can be used.
Media type is `"mediaType"`	When the specified media type of the task is `mediaType,` then...
No process selected	If process is not assigned to the task, then…
Priority is `"operator" "priority"`	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 `priority,` then...
Process is `"process"`	If the task is assigned to `process,` then...
String `"attribute" equals "stringValue"`	When the value of a specified custom attribute of type `string` is equal to the specified `stringValue,` then…
Task is overdue	When the task is overdue, then...
Task overdue in `"period" "periodType"`	Checks if task will be overdue after a given time interval.

Actions

Table 36 describes Standard Rules Template actions.

Table 36: iWD Standard Rules Template Actions

Action	Explanation
Activate task in "period" "periodType"	Activate the task in the specified period of time, using time unit periodType. The activation date/time is used in the IWDBP business process to determine when the task should be moved from the iWD_Captured queue into the Prioritization routing strategy.
Activate task in "period" working "periodType"	Activate the task in the specified period of time, using time unit periodType, according to the task's business calendar (working days, working hours and so on). The activation date/time is used in the IWDBP business process to determine when the task should be moved from the iWD_Captured queue into the Prioritization routing strategy. A business calendar must be assigned to the rule or in a separate rule action before this action can be used.
Activate task beginning next working day	Activates the task at the beginning of the next business day. A business calendar must be assigned to the rule or in a separate rule action before this action can be used.
Activate task end working day	Activates the task at the end of the working day when the task was received. A business calendar must be assigned to the rule or in a separate rule action before this action can be used.
Archive destination "archive"	Set the archive destination for the task. **Note:** This action has been maintained from iWD 8.0 and 8.1.0 but it is no longer necessary to use it, due to changes in the way task archiving works in the out-of-box iWD business process (IWDBP) in iWD 8.1.1. See “Task Archiving” on page 21 for more information.
Assign business calendar "businessCalendar"	Assign the specified businessCalendar to the task.
Assign distribution point "distributionPoint"	Assign the specified distributionPoint to the task. This rule condition requires that the distributionPoint rule parameter be properly configured so that GRAT can access the list of distribution points from the iWD configuration database. See “Enabling Genesys Rules Authoring Tool to Retrieve Dynamic Data from iWD Configuration Database” on page 83.

Table 36: iWD Standard Rules Template Actions (Continued)

Action	Explanation
Assign iWD department "department"	Assign the task to the specified department.
Assign iWD process "process"	Assign the task to the specified process.
Assign WFM Activity "wfmActivity"	Assign the specified Genesys Workforce Management Activity called wfmActivity to the task. To ensure GRS can read objects from Genesys Workforce Management, you must configure the following object in the Configuration Manager under specific tenant's Scripts folder: Name—WFM Connection (or other descriptive name) Type— Data Collection Under Annex tab, configure the following section: Section—wfm Under this section, configure the following properties: wfmCfgServerApplName—CME application name of the WFM Server wfmCfgServerUserName—User name that is used to connect to WFM Server wfmCfgServerPassword—Password that is used to connect to WFM Server wfmServerUrl—URL that is used to connect to WFM Server. The URL must have the following format: http://<wfm_server_host_name>:<wfm_server_port_number> For example, if the WFM Server is running on a host named WFMHost001 on port 7023, the URL would be: http://wfmhost001:7023
Increase iWD priority "priority"	Increase the iWD priority of the task by the specified amount defined in priority.
Multiply business value "multiplier"	Multiply the business value of the task by the specified multiplier.
Reprioritize after "period" "periodType"	Reapply prioritization rules to the task after the specified period of time, using time unit periodType from the current time.
Reprioritize after "period" working "periodType"	Reapply prioritization rules to the task after the specified period of time, using time unit periodType, according to the task's business calendar (working days, working hours, and so on). A business calendar must be assigned to the rule package, or directly to the task, before this action can be used.

Table 36: iWD Standard Rules Template Actions (Continued)

Action	Explanation
Request agent "agent"	Request a specific agent for the task.
Request skill "skill"	Request a specific skill for the task.
Request agent group "agentGroup"	Specify which agentGroup is required to process the task.
Request place group "placeGroup"	Specify which placeGroup is required to process the task.
Set activation date from "customAttribute"	Set the activation date/time of the task from the specified customAttribute of the task. The activation date/time is used in the IWDBP business process to determine when the task should be moved from the iWD_Captured queue into the Prioritization routing strategy.
Set activation time "time"	Set the time when the task will be activated. The activation date/time is used in the IWDBP business process to determine when the task should be moved from the iWD_Captured queue into the Prioritization routing strategy. The value for "time" must be entered by the rules author in UTC, because it will be attached to the task in the IWD_activationDateTime attribute. The value of this attribute will be displayed to users in the Global Task List based on their operating system's time zone settings. Entering time values in UTC is not intuitive for a business user, therefore, if there is a business reason to attach a specific activation time to a task, you might want to create a custom rule condition that displays the local time (considering the time zone of the business, for example) in the Language Expression, while assigning the corresponding value in UTC, in the Rule Language Mapping. Take care with date changes when specifying the activation time in a local time zone. For example, 23:00 in GMT-5 is 04:00 in UTC on the next day, so date should also be incremented by one day.
Set business value "businessValue"	Set business value of the task to the specified businessValue.
Set due date from "customAttribute"	Set the due date/time of the task from the specified customAttribute of the task.

Table 36: iWD Standard Rules Template Actions (Continued)

Action	Explanation
Set due time "time"	Set the time when the task is due. The value for "time" must be entered by the rules author in UTC, because it will be attached to the task in the IWD_dueDateTime attribute. The value of this attribute will be displayed to users in the Global Task List, based on their operating system's time zone settings. Entering time values in UTC is not intuitive for a business user, therefore, if there is a business reason to attach a specific due time to a task, you might want to create a custom rule condition that displays the local time (considering the time zone of the business, for example) in the Language Expression, while assigning the corresponding value in UTC, in the Rule Language Mapping. Take care with date changes when specifying the activation time in a local time zone. For example, 23:00 in GMT-5 is 04:00 in UTC on the next day, so date should also be incremented by one day.
Set integer "attribute" value "integerValue"	Set the value of a specified custom attribute of type integer to the specified integerValue.
Set priority "priority"	Set the iWD priority of the task to the specified value priority.
Set string "attribute" value "stringValue"	Set the value of a specified custom attribute of type string to the specified stringValue.
Task Due in "period" "periodType"	The task is due after the specified period of time, using time unit periodType.
Task Due in "period" working "periodType"	The task is due after the specified period of time, using time unit periodType, according to the task's business calendar (working days, working hours, and so on). A business calendar must be assigned to the rule package, or directly to the task, before this action can be used.
Task expires in "period" "periodType"	Sets the task's expiration date/time after the specified period of time, using time unit periodType. A task's expiration date/time is what is used in the IWDBP business process to determine when a task should be deleted from the Interaction Server database.

Table 36: iWD Standard Rules Template Actions (Continued)

Action	Explanation
Task expires in "`period`" working "`periodType`"	Sets the task's expiration date/time after the specified period of time, using time unit `periodType`, according to the task's business calendar (working days, working hours, and so on). A task's expiration date/time is what is used in the IWDBP business process to determine when a task should be task should be deleted from the Interaction Server database. **Note:** A business calendar must be assigned to the rule package or directly to the task, before this action can be used.

Parameters

Table 37 describes parameters for the Standard Rules Template.

Table 37: iWD Standard Rules Template Parameters

Parameter	Description
`agent`	Presents you with a list of agents' user names that are read from the Genesys Configuration Server database. This list is dynamic; it changes as agents are added or removed. The agents displayed will be based on the access control of the Genesys Rules Authoring Tool user.
`agentGroup`	Presents you with a list of Agent Groups that are read from the Genesys Configuration Server database. This list is dynamic; it changes as agents are added or removed. The agent groups displayed will be based on the access control of the Genesys Rules Authoring Tool user.
`archive`	Presents you with an enumerated list of possible archive destinations for expired tasks. This parameter uses an `Enum`, which is included in the Standard Rules Template, called `archive`. **Note:** This action has been maintained from iWD 8.0 and 8.1.0 but it is no longer necessary to use it, due to changes in the way task archiving works in the out-of-box iWD business process (IWDBP) in iWD 8.1.1. See “Task Archiving” on page 21 for more information.
`attribute`	Enables you to enter text that represents the name of a task attribute.

Table 37: iWD Standard Rules Template Parameters (Continued)

Parameter	Description
businessCalendar	Presents you with a list of business calendars that are defined in the rule package. This list is dynamic; it changes as business calendars are added or removed.
businessValue	Enables you to enter the numeric value that represents business value.
businessValue_From	Enables you to enter a numeric value that represents the lower boundary of business value.
businessValue_To	Enables you to enter a numeric value that represents the upper boundary of business value.
capturePoint	Presents you with a list of Capture Points that is generated dynamically by reading the iWD configuration database.
customAttribute	Enables you to enter text that represents the name of a task custom attribute.
department	Presents you with a list of departments that are defined in the iWD Solution. This list is dynamic; it changes as departments are added or removed.
distributionPoint	Presents you with a list of distribution points that are defined in the iWD Solution. This list is dynamic; it changes based on entries in the distributionPoints lookup table. See “Lookup Tables” on page 156.
diffByPeriod	Returns the difference between the two dates in the specified time units (minutes, hours, days).
difWorkingDays	Returns the difference between two dates in working days.
difWorkingHours	Returns the difference between two dates in working hours.
difWorkingMinutes	Returns the difference between two dates in working minutes.
integerValue	Enables you to enter an integer value for use in rule conditions or actions that evaluate the value of task custom attributes.

Table 37: iWD Standard Rules Template Parameters (Continued)

Parameter	Description
mediaType	Presents you with a list of media types that are defined in the Genesys Configuration Server under the tenant. This list is dynamic; it changes as media type attributes get added or removed.
multiplier	Enables you enter to enter a numeric value by which some other parameter (such as priority) will be multiplied in a rule.
operator	Provides an enumerated list of operators equal to, not equal to, less than, less than or equal to, greater than, and greater than or equal to, that can be used to compare values of custom attributes of type integer, to values specified in a rule condition. This parameter uses an Enum, which is included in the Standard Rules Template, called operator.
period	Enables you to enter a numeric value. Combined with the period type, it gives the actual value of the time period.
periodFrom	Enables you to enter a numeric value that represents the start point of a period, in time units, according to period type.
periodTo	Enables you to enter the numeric value that represents the end point of a period, in time units, according to period type.
periodType	Presents an enumerated list of types of time periods, such as days, hours or minutes. This parameter uses an Enum, which is included in the Standard Rules Template, called periodType.
placeGroup	Presents you with a list of place group names that are read from the Genesys Configuration Server database. This list is dynamic; it changes as place groups are added or removed. The place groups displayed will be based on the access control of the Genesys Rules Authoring Tool user.
priority	Enables you to enter the numeric value that represents iWD priority.

Table 37: iWD Standard Rules Template Parameters (Continued)

Parameter	Description
`process`	Presents you with a list of processes that are defined in the iWD Solution. This list is dynamic; it changes as processes are added or removed.
`skill`	Presents you with a list of skills that are read from the Genesys Configuration Server database. This list is dynamic; it changes as skills are added or removed. The skills displayed will be based on the access control of the Genesys Rules Authoring Tool user.
`stringValue`	Enables you to enter a string value for use in rule conditions or actions that evaluate the value of task custom attributes.
`taskChannels`	Presents the user with a list of task channels, read from an iWD Lookup Table (see page 156).
`time`	Enables you to enter a time value that is used in various rule conditions and actions.
`wfmActivity`	Presents a list of WFM activities, retrieved dynamically from the WFM Server.

Functions

Table 38describes functions for the Standard Rules Template.

Table 38: iWD Standard Rules Template Functions

Function	Description
`adjustWorkingDate`	Adds or subtracts a given time interval from the given date according to the current business calendar.
`compareDate`	Compares the value of two dates. For a list of possible returned values, see comments within the function.
`compareInteger`	Compares the value of two integers, taking as inputs the two integers and a comparison operator. Returns `true` or `false`.
`getCurrentCalendar`	Returns the current business calendar ID.

Table 38: iWD Standard Rules Template Functions

Function	Description
`getCurrentDT`	Returns the current date and time, in UTC.
`getDTValue`	Gets the value of a task attribute as date/time.
`getIntValue`	Gets the value of a task attribute as an integer.
`getPeriodDTFrom`	Adds or subtracts a given time interval from the given date, regardless of the business calendar.
`getStringValue`	Gets the value of a task attribute as a string.
`getWFMActivity`	Retrieves WFM Activity.
`getWFMActivityContext`	Retrieves WFM Activity context (Business Unit or Site).
`increaseIntegerValue`	Increases the value of an integer by adding to it.
`isNull`	Is `true` if a given task attribute value is `Null`, otherwise, the value is `false`.
`isProcess`	Is `true` if a given task is assigned to a given process, otherwise, the value is `false`.
`isWorkingDay`	Returns `true` if the current day is the working day.
`isWorkingTime`	Returns `true` if the current time is the working time.
`multiplyIntegerValue`	Increases the value of an integer by multiplying it by some factor.
`notNull`	Is `true` if a task attribute value is not `Null`, otherwise, the value is `false`.
`setDepartment`	Sets the department for a task.
`setDepartmentAndProcess`	Sets the department and process of the given task from a given string, in `department|process` format.

Table 38: iWD Standard Rules Template Functions

Function	Description
`setDTValue`	Sets the value of a task attribute as date/time.
`setIntValue`	Sets the value of a task attribute as an integer.
`setProcess`	Sets the process for a task.
`setStringValue`	Sets the value of a task attribute as a string.
`setTime`	Sets the time portion of a given date/time value from a string value, in `hh:mm` format.

Rule Authoring

Rule authoring for iWD is done through the Genesys Rules Authoring Tool (GRAT). This section describes how to log into the Genesys Rules Authoring Tool, some general information about its usage for iWD, and how to use it for creating decision tables, linear rules, and business calendars.

Changes in Archiving in iWD Release 8.1.1

A new archiving solution is implemented in iWD release 8.1.1. Pre-8.1.1 , archive rules (the rules that are applied during the archiving phase) are no longer applicable.

The iWD Standard Rules Template still contains a dedicated phase called `Archiving`. This is done for compatibility reasons to allow migration from earlier versions of the iWD, where archiving rules may have been used. After migration, users should review the migrated rules and adjust them according to the new archiving strategy.

Logging into the Genesys Rules Authoring Tool

The Genesys Rules Authoring Tool is a web browser-based user interface that you can log into directly by navigating to a URL and entering a user name and password (for instructions, see the *Genesys Rules System 8.1 Deployment Guide*) or you can launch the tool and log in by using the `Rules Authoring` link on the iWD Manager navigation bar. To ensure that a user can launch the Genesys Rules Authoring Tool from iWD Manager, see Procedure:

Configuring iWD tenant and user for Rules Authoring Tool access, on page 230.

Note: The User permissions to the various capabilities of the Genesys Rules Authoring Tool are controlled by Role-Based Access Control, which is configured through Genesys Administrator (if you are not using Genesys Administrator, you will have full access to the Genesys Rules Authoring Tool). See “Role-Based Access Control”, Chapter 5 in the *Genesys Rules System 8.1 Deployment Guide.*

Procedure: Configuring iWD tenant and user for Rules Authoring Tool access

Purpose: To complete the necessary configuration of an iWD managed tenant and an iWD user's security role, so that the Rules Authoring link appears on the iWD Manager navigation bar, properly launch the Genesys Rules Authoring Tool web application, and log in the user.

Prerequisites

- iWD Manager and the Genesys Rules Authoring Tool are installed and configured.

Start of procedure

1. Log in to iWD Manager as a user with full administrative privileges
2. From the `tenant` drop-down list, select your iWD-managed tenant.
3. Select the `Profile` link.
4. Under Genesys Rules Authoring Tool URL, enter a valid URL for the Genesys Rules Authoring Tool.
5. Test this URL in a separate browser window to ensure that it brings up the login screen.

   The default syntax for the URL should be:

   `http://<host>:<port>/genesys-rules-authoring`

   where;

   - `<host>`—Is your application server host name.
   - `<port>`—Is the listening port of your application server (usually `8080` by default).

- genesys-rules-authoring—is the name of the Genesys Rules Authoring Tool web application that is deployed on your application server. (This will be the default name of that web application unless you have modified it.)

6. Click Save.
7. Select the Security Policy link.
8. Select the name of an existing Security Role, to which you want to give access to the Genesys Rules Authoring Tool, or create a new one.
9. Scroll down to the Application Permissions section.
10. Under the Run column, check the checkbox for the Rules Authoring permission.
11. Click Save.

End of procedure

General Overview of Genesys Rules Authoring Tool for iWD Users

When you launch the Genesys Rules Authoring Tool from iWD Manager, the active tenant will be the Configuration Server tenant that is linked to the iWD managed tenant you were working with in iWD Manager.

The GRAT navigation tree will display the iWD Solutions that are configured under the tenant. If you don't see any Solutions in the navigation tree, but you have one or more Solutions configured in iWD Manager, you must ensure that the user has permission to the appropriate folders under the Business Structure folder in Configuration Server. See Procedure: Configuring Genesys Rules Authoring Tool User Access Control to iWD Business Structure, on page 232.

Note: The Business Structure folder is created in either of the following scenarios:

- During the migration process from an earlier version of iWD to iWD 8.1.
- If the iWD business structure has been pushed to the Genesys Rules System from iWD Manager.

See Procedure: Push iWD Business Structure to the Genesys Rules System, on page 233.

Procedure: Configuring Genesys Rules Authoring Tool User Access Control to iWD Business Structure

Purpose: To configure access control for the Genesys Rules Authoring Tool user, to allow him/her to see the appropriate iWD Solutions when logged into GRAT.

Prerequisites

- iWD Manager and the Genesys Rules Authoring Tool are installed and configured.

Start of procedure

1. Log in to Genesys Administrator or Genesys Configuration Manager.
2. Navigate to the tenant that maps to the iWD managed tenant with which you are working.
3. Locate the `Business Structure` folder (in Genesys Administrator, go to `Environment > Business Units/Sites`).
4. In the `Business Structure` folder, locate the iWD Solution folder to which you want the Genesys Rules Authoring Tool user to have access.
5. If you are using Genesys Configuration Server:
   - On the `Security` tab of that iWD Solution, configure at least `Read` access for the Genesys Rules Authoring Tool user, or for at least one Configuration Server Access Group to which that user belongs.

   If you are using Genesys Administrator:

   a. Highlight the `iWD Solution` folder and select `Edit`.

   b. Select the `Permissions` tab.

   Use either `Add Access Group` or `Add User` to configure at least `Read` access to this iWD Solution, for the necessary Genesys Rules Authoring Tool user or his Access Group(s).

   Note: When you add the user or Access Group, you can decide whether or not to propagate the permissions. If you do not propagate the permissions, you will need to explicitly add permissions to the `Department(s) and Process(es)` under the iWD Solution, to which you want the user or access group to have access.

6. To grant access to any additional iWD Solutions, repeat Steps 4 to 6.

End of procedure

Procedure: Push iWD Business Structure to the Genesys Rules System

Purpose: To synchronize the iWD business structure that has been created in iWD Manager with Genesys Configuration Server.

Summary

After you complete this procedure, the business structure will become available to the Genesys Rules System, specifically to the Genesys Rules Authoring Tool so it can display the appropriate business structure for use with iWD.

Prerequisites

- iWD Manager and the Genesys Rules Authoring Tool are installed and configured
- Some business structure (iWD Solution, Departments, and/or Processes) is created in iWD Manager.

Start of procedure

1. Log in to iWD Manager.
2. Click `Departments and Processes`.
3. Expand the iWD Solution you want to push to the Genesys Rules System, and select `Push to Rules System`.

   Alternatively, if you have not already performed this action, you will see the following notification on the top of the iWD Manager screen informing you that “There are changes to be pushed to Rules System: [Solution Name]”, with a hyperlink that will take you to the appropriate screen where you can execute the push action.
4. At the bottom of the `Push to Rules System` screen, click `Execute`.

   In the `Messages` pane, you will see one or more messages indicating the success or failure of the push action.

Note: If the action fails, it might be because the user does not have the appropriate permissions to the *parent* folder. For example, if you have just added a new Process under an existing Department, and the iWD Manager user does not have the appropriate permission (`Full Control`) to that Department business structure folder in Configuration Server, you will not be able to create the new Process.

End of procedure

Next Steps

- Create a rule package in Genesys Rules Authoring Tool that you can use to create business rules. See Procedure: Creating a New Rule Package.

If you have successfully created your business structure in iWD Manager and pushed it to the Genesys Rules System, you are ready to create a rule package in Genesys Rules Authoring Tool, to start creating business rules.

Tip: Prior to release 8.1.1, if you delete a Process or Department in iWD Manager, you can also push these changes to the Genesys Rules System. However, if you delete an entire Solution in iWD Manager, you cannot push these changes to the Genesys Rules System through iWD Manager. This is because the `Push to Rules System` action is only visible in the navigation tree under the Solution. In this case, you must manually delete the `Solution` folder under `Business Structure,` through Genesys Administrator or Genesys Configuration Manager.

Until you manually delete the Solution folder by using Genesys Administrator or Genesys Configuration Manager, you will continue to see this unwanted Solution in the Genesys Rules Authoring Tool.

Also, to delete a parent folder in Configuration Manager or Genesys Administrator, such as an iWD Solution, you must first delete the child folders, such as the Departments. If there are Processes under those Departments, those must be deleted first.

After you have a working environment, from time to time you might need to modify the business structure in iWD Manager. After making those modifications, you must the following steps in this order:

1. Push the business structure changes to the Genesys Rules System, as described in this section.
2. Modify your business rules, if necessary, in GRAT.
3. Deploy your rule package in GRAT.
4. Deploy your iWD Solution in iWD Manager.

From release 8.1.1 onwards, if you delete a solution, you will be prompted to select whether you want to delete it from the rules system as well. If you confirm the deletion, the solution with all business objects will be removed. If there was a rules package deployed from that solution, you need to first delete the rules package manually.

Procedure:
Creating a New Rule Package

Purpose: To create a new rule package in Genesys Rules Authoring Tool.

Summary

The rule package is the parent object for all the business rules for an iWD Solution.

Start of procedure

1. Log in to iWD Manager and launch the Genesys Rules Authoring Tool.

2. In the navigation tree, expand the Solution with which you want to work.

3. Select `New Rule Package.`
4. On the right side of the screen, fill in the properties for the rule package, such as.
   - `Package Name`—Used internally and primarily in the Configuration Server List Object that will be read in the iWD business process, so the business process knows which rule package should be evaluated by the Genesys Rules Engine. There are some reserved keywords that must not be used when you name the rule package. See 'Creating Rule Packages" topic in the *Genesys Rules Authoring 8.1 Help.*
   - `Business Name`—Enter any user-friendly name you wish to provide to identify the rule package.
   - `Package Type`—Select `iWD.` Selecting iWD will ensure that the list of available rule templates of type `iWD` will be displayed.
   - `Rule Package` (Optional)—Enter a description for the rule package.
   - `Template` section—Select one or more rule templates. This will determine which collection of rule actions and conditions will be made available to the business rules authors who are responsible for creating rules in this package.
5. Save the rule package.

   The new rule package will appear in the navigation tree on the left side of the pane, displaying all the Departments and Processes underneath the package, based on the access control of the user.

End of procedure

Next Steps

- After a rule package has been created, you can start creating business rules. For the descriptions of the various types of rules and rule objects, see "Rules Overview" on page 237), "Decision Tables" on page 239, "Linear Rules" on page 238, and "Business Calendars" on page 241.

Note: The list of rule templates that are available to the person creating the rule package will depend on that user's access to the `Script` objects that represent each template. In Configuration Server, under each tenant, there is a `Script` folder that contains a subfolder called `Template Access Control.` In that folder, there is a script of type `Data Collection,` for each rule template published to the rules repository. The access control defined on the `Security` tab of that `Script` object will determine which users and access groups can use that template in a rule package.

Rules Overview

In the Genesys Rules Authoring Tool, there are three levels at which business rules can be created:

- Rule Package (referred to as Global Rules)
- Department
- Process

When the appropriate node is selected on the rule package tree, you can then select the `Rules` tab to view or edit the rules for that level of the business structure. Rules are presented in a list, with an associated phase. The order of the rules is relevant, because they will be evaluated, within a particular phase, in the same order as they appear. 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 iWD extended or custom attribute can be read or updated by business rule conditions or actions, respectively.

Global Rules

Rules that are created at the rule-package level are also known as *Global Rules*. Global rules enable you to specify rules that will apply to the entire iWD Solution. For example, they enable you to configure rules that classify or prioritize all tasks globally, instead of at a lower level of the business structure. Global rules are applied before any other rules.

This means that each rule phase (classification and prioritization) is triggered from within the IWDBP business process in the following sequence:

1. Global rules
2. Department rules
3. Process rules

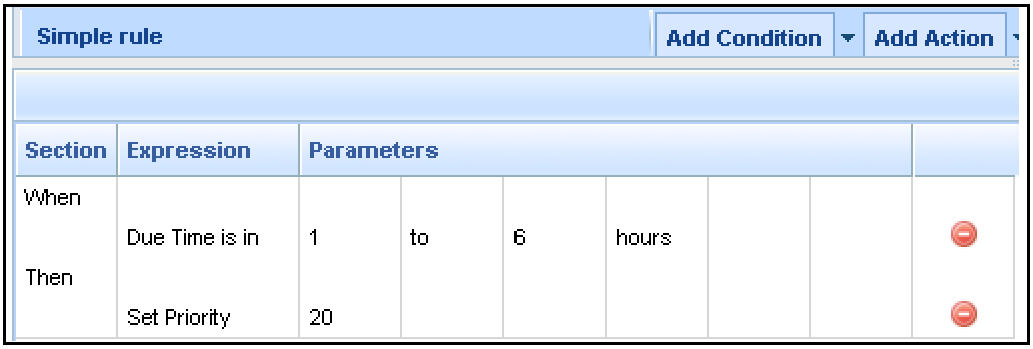
Note: 'A classification rule must be configured that assigns the task to a Process, or the Process can be assigned directly in the `createTask` message when the task is captured by a capture point. If a Process is not assigned to the task in either of these two methods, the task status will be changed to `Error Held.` For example, a department can be assigned in a global rule, followed by assigning a process at the department level. Alternatively, a process might be directly assigned in a global rule. If a process is assigned in both global rules and department rules, based on the same conditions, the department rules will override.

Capture Point Rules

In iWD 8.1. you can use the rule condition `Capture Point is` from the iWD Standard Rules Template, at the Global Rules level. Classification rules are no longer created at the Capture Point level (as they were in iWD 7.6.1 and 8.0).

Linear Rules

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. See also Figure 64 on page 238. See Procedure: Specifying a linear rule, on page 238, which describes how to specify a linear rule.



Section	Expression	Parameters						
When								
	Due Time is in	1	to	6	hours			⊖
Then								
	Set Priority	20						⊖

Figure 64: Linear Rule

Procedure: Specifying a linear rule

Purpose: To specify a linear rule for a simple business case.

Start of procedure

1. In Genesys Rules Authoring Tool, expand the rule package and select a node at which you want to create the rule.
2. On the right side of Genesys Rules Authoring Tool panel, on the `Rules` tab, click `New Linear Rule.`
3. Enter a Name for the rule that identifies it.
4. Optionally, enter a Description for the rule.
5. Select the `Phase` in which to apply the rule.

   You can choose `classification` or `prioritization.`
6. Optionally, assign a `Calendar` (business calendar) to the rule.

7. If required, set the Start Date and End Date.

   Note: If Start Date and End Date are left empty the rule activation period is unconstrained.

8. From the Add Condition combo box, select one or several conditions for the rule.

   By default, conditions are concatenated by using the Logical AND operator. However, you can select other functions (for example, “or" or “and not") from the Add Condition list, and you can also select multiple conditions and use the Group (or Ungroup) functions.

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. Genesys strongly recommends using a custom rule package for all new rule actions or conditions.

9. Select one or several actions for the rule from the Add Action combo box.
10. To save the specified rule, click Save.

    You can optionally enter a check-in comment, which will be available later on the Audit Trail tab.

End of procedure

Next Steps

After you have created a linear rule, you can create additional linear rules or decision tables, or deploy your rule package. “Rule Deployment” on page 244.

Tip: The available conditions and actions that are presented to the rules author is driven by the set of rule templates to which the user has access. This depends on the user's permissions to the Script objects that represent each template. In Configuration Server, under each tenant, there is a Script folder that contains a subfolder called Template Access Control. In that folder, there is a script of type Data Collection, for each rule template published to the rules repository. The access control that is defined on the Security tab of that Script object will determine which users can access which rule templates, which will control the final list of rule actions and conditions presented.

Decision Tables

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. See Procedure: Defining a decision table, on page 240, which describes how to define a decision table.

Procedure: Defining a decision table

Purpose: To specify a decision table rule for a complex business case.

Start of procedure

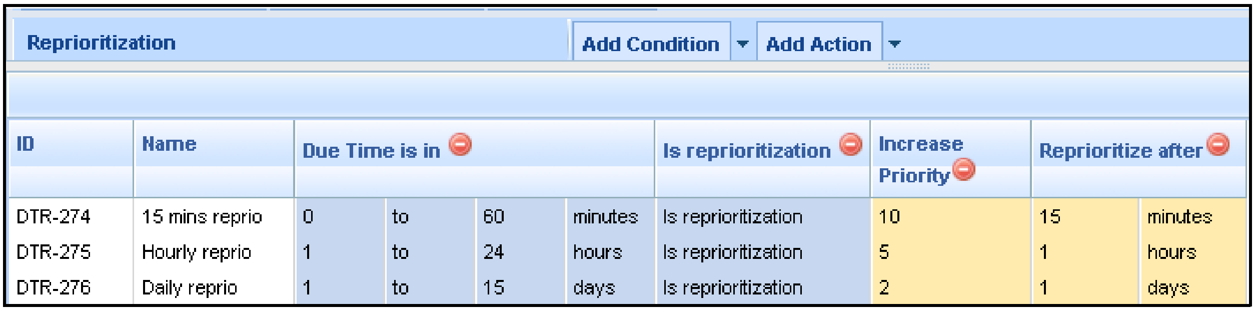
1. On the `Rules` tab, click `New Decision Table`.
2. Specify the rule `Name`, `Description`, `Phase`, `Calendar`, `Start Date`, and `End Date`, if required (as you did in Steps 3 to 7 in the Procedure: Specifying a linear rule.)
3. From the `Add Condition` combo box, select zero or more conditions for the rule.
4. From the `Add Action` combo box, select one or more actions for the rule.
5. To add a new row to the decision table, select the green plus (+) icon to the right of the right-most rule action.
6. Enter the required parameters for each rule condition and action.
7. Optionally, enter a `Name` for the row.
8. Repeat Steps 5 through 7, 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.

   Note: To delete a 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).

9. To save the specified rule, click `Save`.
10. Optionally, enter a check-in comment, which will be available later on the `Audit Trail` tab.

    For an example of a decision table, see Figure 65 on page 241.



ID	Name	Due Time is in				Is reprioritization	Increase Priority	Reprioritize after	
DTR-274	15 mins reprio	0	to	60	minutes	Is reprioritization	10	15	minutes
DTR-275	Hourly reprio	1	to	24	hours	Is reprioritization	5	1	hours
DTR-276	Daily reprio	1	to	15	days	Is reprioritization	2	1	days

Figure 65: Decision Table

End of procedure

Next Steps

- After you have created a decision table, you can create additional decision tables or linear rules, or deploy your rule package. See “Rule Deployment” on page 244.

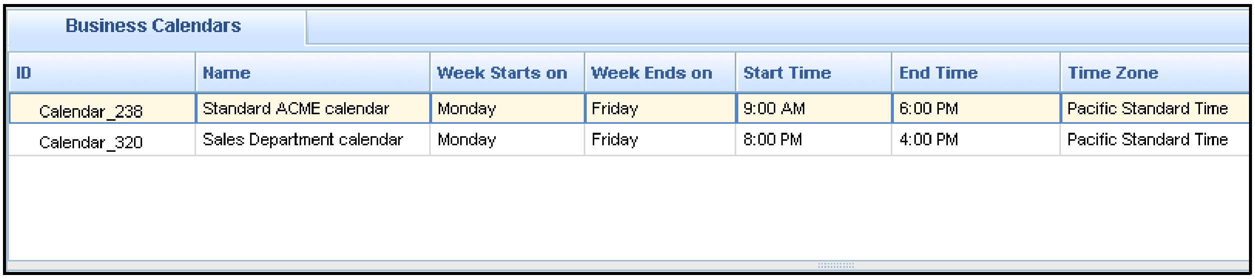
Tip: The available conditions and actions that are presented to the rules author is driven by the set of rule templates to which the user has access. This depends on the user's permissions to the `Script` objects that represent each template. In Configuration Server, under each tenant, there is a `Script` folder that contains a subfolder called `Template Access Control`. In that folder, there is a script of type `Data Collection`, for each rule template published to the rules repository. The access control that is defined on the `Security` tab of that `Script` object will determine which users can access which rule templates, which will control the final list of rule actions and conditions presented.

Business Calendars

A business calendar is a set of rules that define working days and hours, and 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 can be assigned once (for example, at the Global Rule level), or can be assigned dynamically in a rule when needed.

Managing Business Calendars in the Genesys Rules Authoring Tool

To create new (or to manage existing) business calendars, launch the Genesys Rules Authoring Tool and expand a rule package. Select the Business Calendars entry in the tree. An example of a rule package's business calendars is shown in Figure 66.

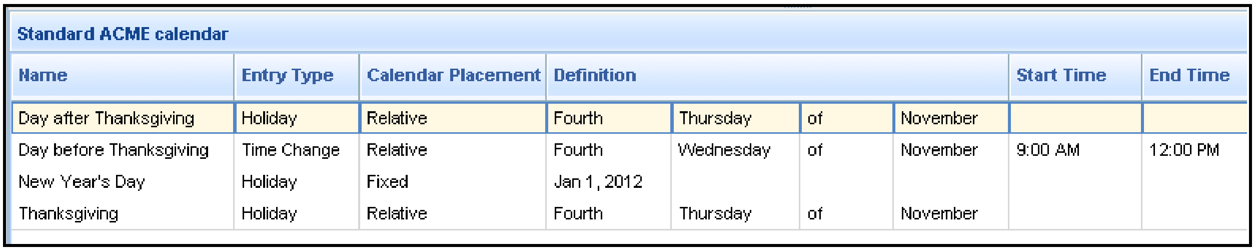


Business Calendars

ID	Name	Week Starts on	Week Ends on	Start Time	End Time	Time Zone
Calendar_238	Standard ACME calendar	Monday	Friday	9:00 AM	6:00 PM	Pacific Standard Time
Calendar_320	Sales Department calendar	Monday	Friday	8:00 PM	4:00 PM	Pacific Standard Time

Figure 66: List of Solution's Business Calendar

The right side of the Genesys Rules Authoring Tool is split horizontally into two panes. The upper displays a list of a rule package's business calendars. The `New Calendar` button that is below 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 side of the business calendar in the business calendar list. When a calendar list is selected, the lower pane displays the attributes of the selected business calendar. Figure 67 shows an example of the attributes of a business calendar.



Standard ACME calendar

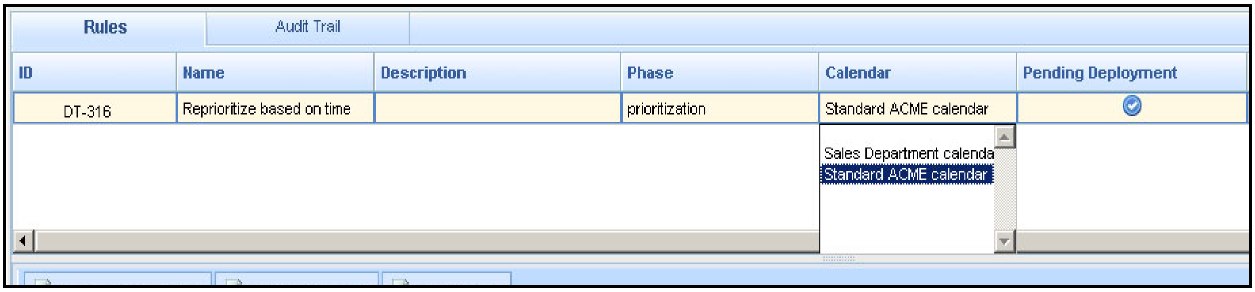
Name	Entry Type	Calendar Placement	Definition				Start Time	End Time
Day after Thanksgiving	Holiday	Relative	Fourth	Thursday	of	November		
Day before Thanksgiving	Time Change	Relative	Fourth	Wednesday	of	November	9:00 AM	12:00 PM
New Year's Day	Holiday	Fixed	Jan 1, 2012					
Thanksgiving	Holiday	Relative	Fourth	Thursday	of	November		

Figure 67: Business Calendar Attributes

Business calendars consist of a set of standard mandatory attributes and optional business calendar rules. See "Business Calendars" in the *Genesys Rules Authoring Tool 8.1 Help*.

Using Business Calendars in iWD Rules

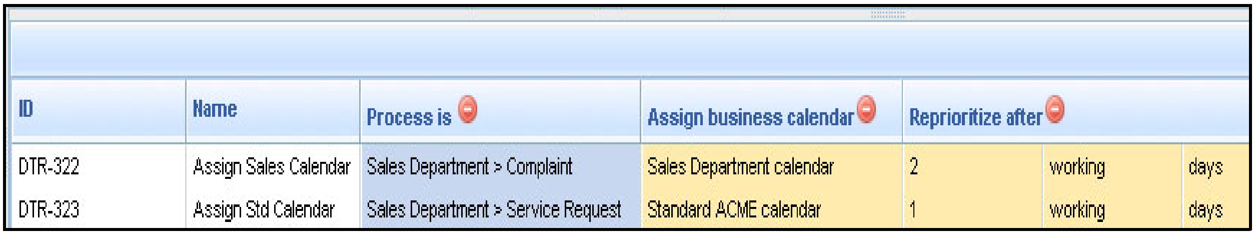
After business calendars are defined, you can use them in rules. Business calendars must be assigned to a task before any business calendar related calculations can be performed on task values. A Calendar can be assigned at the parent rule level, as show in Figure 68, or to an individual rule as shown in Figure 69. Only one calendar can be assigned to a task at a time, so a calendar can be assigned by one rule and then, overwritten by a later rule.



ID	Name	Description	Phase	Calendar	Pending Deployment
DT-316	Reprioritize based on time		prioritization	Standard ACME calendar	✓

Sales Department calenda
Standard ACME calendar

Figure 68: Assigning a Business Calendar to a Parent Rule



ID	Name	Process is	Assign business calendar	Reprioritize after		
DTR-322	Assign Sales Calendar	Sales Department > Complaint	Sales Department calendar	2	working	days
DTR-323	Assign Std Calendar	Sales Department > Service Request	Standard ACME calendar	1	working	days

Figure 69: Assigning a Business Calendar Dynamically in a Rule

Table 39 describes the business calendar-related actions that are available in the iWD Standard Rules Template.

Table 39: iWD Rules Conditions and Actions Using Business Calendars

Condition/ 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 drop-down list displays a list of business calendars that are defined for the rule package.
Is Working Day	N/A	Calculates whether the current date/time is a working day, according to the assigned business calendar.
Is Working Time	N/A	Calculates whether the current date/time is working time, according to the assigned business calendar.
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 39: iWD Rules Conditions and Actions Using Business Calendars

Condition/ 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 the standard rule actions that use business calendars, you can build other rule expressions that use business calendar functions. For more information about how to build these other rule expressions, see https://sites.google.com/a/iwdlab.com/iwd8/rules/bc.

Rule Deployment

Once you have created all the necessary rules in a rule package, it is time to deploy the rule package to an instance of the Genesys Rules Engine. Once the package is deployed, it can be invoked by a client such as the iWD business process (IWDBP).

Rule package deployment is done through the Genesys Rules Authoring Tool, and is independent of the iWD Solution deployment in iWD Manager. Thus, you can deploy a new version of a rule package for an iWD Solution without having to redeploy the entire Solution, (as was the case in iWD 7.6.1 and 8.0).

When a rule is created or edited and it has not been deployed, there is a checkmark in the `Pending Deployment` column of the rule. See Figure 69.

In addition, when a rule package requires deployment or redeployment, there will be a visual indication next to the `Deploy Rules` node in the `Genesys Rules Authoring Tool` navigation tree, under the rule package itself. See Figure 70.

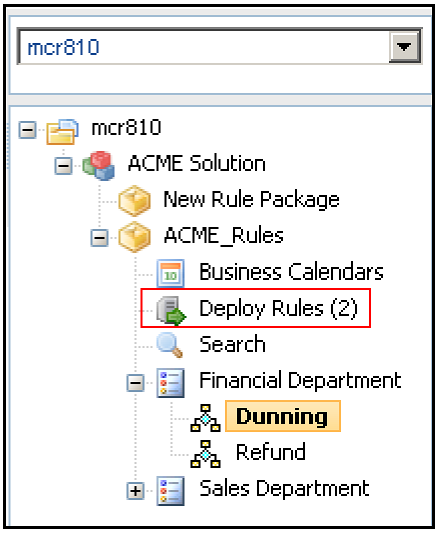


Figure 70: Deploy Rules Node

For more information about rule package deployment, including scheduled deployment and deployment history, see the *Genesys Rules Authoring Tool 8.1 Help*.

Rule Evaluation

After a rule package has been deployed to a Genesys Rules Engine, it can be invoked by a client. The standard client that invokes the Genesys Rules Engine is the iWD business process (IWDBP). For information about other ways to invoke rule packages, see the *Genesys Rules System 8.1 Deployment Guide*.

It is important to understand how rules are evaluated, when an evaluation is requested from the IWDBP business process.

IWDBP has the following two routing strategies that invoke rule evaluation from the Genesys Rules Engine.

1. Classification—Calls the Genesys Rules Engine and requests an evaluation of all rules for the iWD Solution's associated rule package that belong to the Classification phase.
2. Prioritization—Calls the Genesys Rules Engine and requests an evaluation of all rules that belong to the Prioritization phase.

Associating the iWD Solution with a Rule Package and Rules Engine

To ensure the IWDBP business process invokes the correct Genesys Rules Engine and the correct rule package, two List Objects must be properly configured: `Iwd_Esp_List` and `Iwd_Package_List`.

Note: Make sure that both of these List Objects are correctly configured, otherwise IWDBP will not work.

Iwd_Esp_List

The IWDBP uses the data from the `Iwd_Esp_List` List Object to correlate the `IWD_SolutionId` to the name of the Business Context Management Service (BCMS) application, and to correlate the `IWD_SolutionId` to the name of the Genesys Rules Engine. The BCMS is the External Service Protocol (ESP) server that the IWDBP business process uses to communicate with the Genesys Rules Engine.

`Iwd_Esp_List` is also used, optionally, to correlate the `IWD_SolutionId` to the name of a Genesys Universal Contact Server (UCS) that is connected to Interaction Server. If this association is configured in `Iwd_Esp_List`, the business logic in IWDBP will update the interaction history in the UCS database to mark the interaction as *done* (the value of the `Status` column in the `Interaction` table in the UCS database will be set to `3`) when the interaction enters one of these queues: `iWD_Completed, iWD_Canceled`, or `iWD_Rejected`.

Iwd_Package_List

The `Iwd_Package_List` List Object is used to correlate the IWD Solution ID (`IWD_SolutionId`) to the name of the rule package that will be evaluated when requests are made to the Genesys Rules Engine from the IWDBP business process.

`Iwd_Package_List` contains a single list called, `RulePackageList`. Create a new key/value pair for each iWD Solution that you have configured under your Configuration Server tenant, where the key or option, is the IWD Solution ID and the value is the Package Name of the rules package.

Note: The user configures two names for a rule package in GRAT. There is the *Package Name* and the *Business Name*. The value that the user enters in the `RulePackageList` list, must reference the Package Name (not the Business Name) of the rule package.

For more information on how to configure these List Objects, see Appendix C, “iWD Business Process (IWDBP),” on page 313.

Rule Evaluation Order

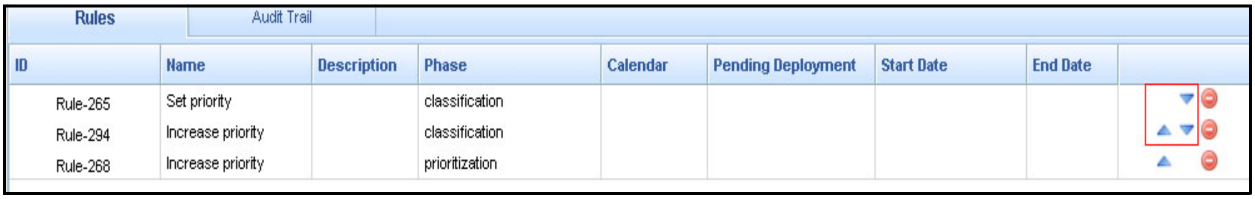
Within all the rules that are configured for a particular phase, the Genesys Rules Engine will evaluate rules in this order:

- Global rules (package-level)
- Department rules
- Process rules

To ensure the Genesys Rules Engine performs its evaluation in this order, the `sequential-mode` option (in the `settings` section in the Genesys Rules Engine `Application`) must be set to `false`. This is the default setting.

To ensure the Genesys Rules Engine evaluates all rules within a particular phase, within a particular node of the business structure hierarchy, the `group-by-level` option (in the `settings` section in the Genesys Rules Authoring Tool server `Application`) must be set to `true`. This is the default setting.

Within the rules that are defined at a particular node of the business structure hierarchy (for example, `Department`), the Genesys Rules Engine will follow the salience (order) that is defined in the rule package. This is controlled by the rules author, who can move rules up and down in the Genesys Rules Authoring Tool. In the example in Figure 71, Rule-265 will be evaluated prior to Rule-294. If the rules author preferred Rule-294 to be evaluated first, he could move that rule up in the evaluation order by using the arrow button on the right side of the rule.



ID	Name	Description	Phase	Calendar	Pending Deployment	Start Date	End Date	
Rule-265	Set priority		classification					▼ ⊖
Rule-294	Increase priority		classification					▲ ▼ ⊖
Rule-268	Increase priority		prioritization					▲ ⊖

Figure 71: Controlling Rule Evaluation Order

Within a set of rules for a particular phase, at a particular node in the business structure hierarchy, an action of one rule can influence a condition in a subsequent rule. To do this, it is necessary to include the `update($data)` parameter; in the `Rule Language Mapping` of the rule action or as a separate rule action. See the following example.

Example: Setting Rules at a Specific Phase

If you had one classification rule at the `Department` level that set the priority of a task to `100` and a second classification rule at the `Department` level that will increase the priority by `90,` but only if the priority has already been set to `100:`

- The first classification rule would look like this:

```
WHEN
    Priority is not equal to 100
THEN
    Set Priority to 100
    Update
```

- The second classification rule would look like this:

```
WHEN
```

```
        Priority is equal to 100
    THEN
        Increase Priority 90
```

In the first classification rule, it is important to specify that the priority be set to `100` only when it is not already equal to `100`. This is to avoid a problem where the Genesys Rules Engine might get into a loop during rule evaluation. Also in the second classification rule, there is a rule action called `Update`. The Rule Language Mapping for this rule action is: `update($data);`

You could also modify the `Rule Language Mapping` of the `Set Priority` rule action, to add `update($data);` to the end of the `Rule Language Mapping`. For example:

```
setIntValue("Priority", {priority}, $data); update ($data);
```

In the latter case, however, you would want to create a new version of the `Set Priority` rule action and give it a different Language Expression to help differentiate it, such as `Set extended Priority`.

Genesys

Chapter

7 Global Task List

This chapter describes how to monitor and manage tasks with intelligent Workload Distribution's (iWD) Global Task List.

This chapter contains the following section:

Task Management

Task Management allows monitoring and management of tasks that are being processed by iWD:

- Use Task Monitoring (see page 249) 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 255) 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 257) 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 72. When a context is selected, the corresponding list of tasks is displayed in the Global Task List.

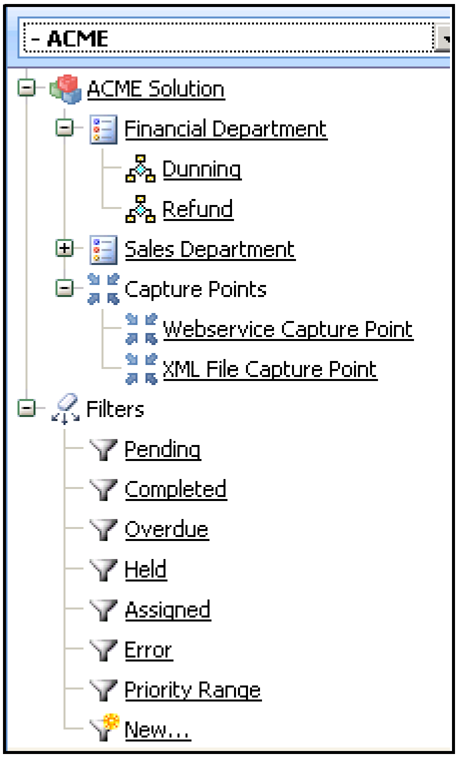
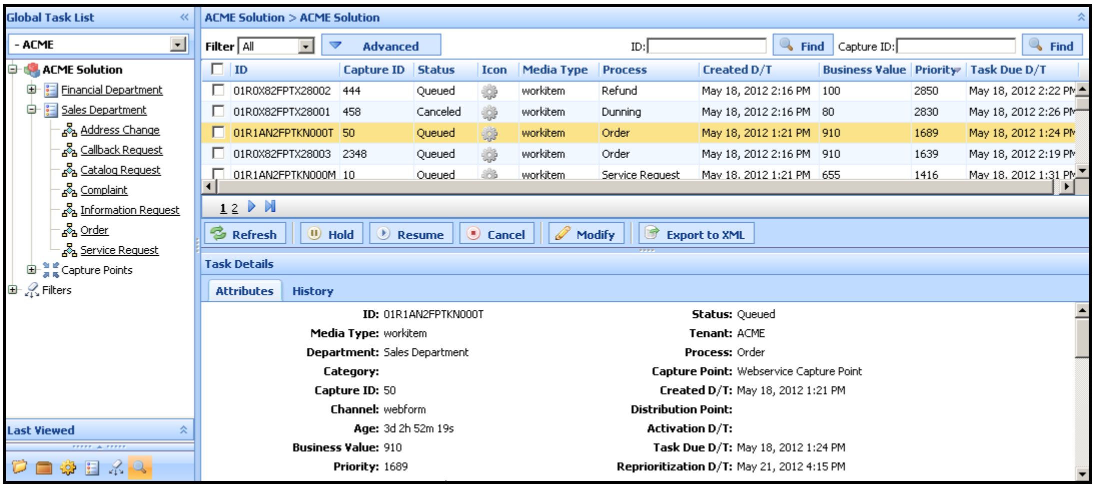


Figure 72: Navigation Tree

Global Task List

The Task List displays a list of tasks for the selected business context, as illustrated below in Figure 73.



☐	ID	Capture ID	Status	Icon	Media Type	Process	Created D/T	Business Value	Priority	Task Due D/T
☐	01R0X82FPTX28002	444	Queued	⚙	workitem	Refund	May 18, 2012 2:16 PM	100	2850	May 18, 2012 2:22 PM
☐	01R0X82FPTX28001	458	Canceled	⚙	workitem	Dunning	May 18, 2012 2:16 PM	80	2830	May 18, 2012 2:26 PM
☐	01R1AN2FPTKN000T	50	Queued	⚙	workitem	Order	May 18, 2012 1:21 PM	910	1689	May 18, 2012 1:24 PM
☐	01R0X82FPTX28003	2348	Queued	⚙	workitem	Order	May 18, 2012 2:16 PM	910	1639	May 18, 2012 2:19 PM
☐	01R1AN2FPTKN000M	10	Queued	⚙	workitem	Service Request	May 18, 2012 1:21 PM	655	1416	May 18, 2012 1:31 PM

ID: 01R1AN2FPTKN000T
Media Type: workitem
Department: Sales Department
Category:
Capture ID: 50
Channel: webform
Age: 3d 2h 52m 19s
Business Value: 910
Priority: 1689

Status: Queued
Tenant: ACME
Process: Order
Capture Point: Webservice Capture Point
Created D/T: May 18, 2012 1:21 PM
Distribution Point:
Activation D/T:
Task Due D/T: May 18, 2012 1:24 PM
Reprioritization D/T: May 21, 2012 4:15 PM

Figure 73: Global Task List

In addition, custom attributes can be made available in the list of Advanced Filters. See “Configuring Custom Attributes” on page 258.

Table 40 lists the actions that are available in the Global Task List.

Table 40: Global Task List Actions

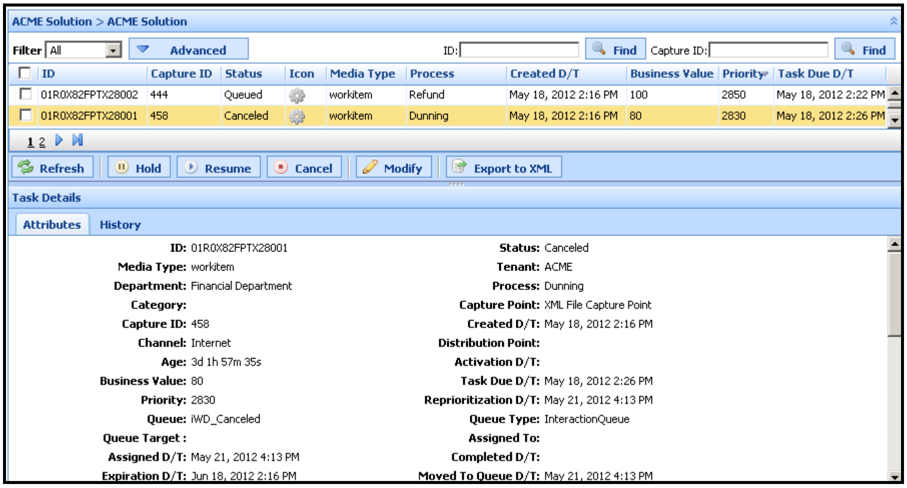
Attribute/Action	Description
Filter	Refines the Global Task List and displayed columns, based on the selected filter. For more information on filters, see “Filters” on page 257.
Advanced Filters	Advanced filters let you further narrow down the tasks that are listed in the Global Task List. Up to three 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.
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 255.

Table 40: Global Task List Actions (Continued)

Attribute/Action	Description
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 73.
Tooltips	Displays the whole value in a tooltip when you hover the mouse over the value in any column of the Global Task List.

Task Details

When a task is selected from the Global Task List, its attributes will be displayed. Figure 74 shows an example of the Task Details.



☐	ID	Capture ID	Status	Icon	Media Type	Process	Created D/T	Business Value	Priority	Task Due D/T
☐	01R0X82FPTX28002	444	Queued		workitem	Refund	May 18, 2012 2:16 PM	100	2850	May 18, 2012 2:22 PM
☐	01R0X82FPTX28001	458	Canceled		workitem	Dunning	May 18, 2012 2:16 PM	80	2830	May 18, 2012 2:26 PM

ID:	01R0X82FPTX28001	Status:	Canceled
Media Type:	workitem	Tenant:	ACME
Department:	Financial Department	Process:	Dunning
Category:		Capture Point:	XML File Capture Point
Capture ID:	458	Created D/T:	May 18, 2012 2:16 PM
Channel:	Internet	Distribution Point:	
Age:	3d 1h 57m 35s	Activation D/T:	
Business Value:	80	Task Due D/T:	May 18, 2012 2:26 PM
Priority:	2830	Reprioritization D/T:	May 21, 2012 4:13 PM
Queue:	iWD_Canceled	Queue Type:	InteractionQueue
Queue Target :		Assigned To:	
Assigned D/T:	May 21, 2012 4:13 PM	Completed D/T:	
Expiration D/T:	Jun 18, 2012 2:16 PM	Moved To Queue D/T:	May 21, 2012 4:13 PM

Figure 74: Task Details

Attributes are broken down into three sections:

- Top—Core task attributes.
- Middle—Extended task attributes. These are 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 185). User-friendly on-screen labels for custom attributes can also be configured. See “Configuring Custom Attributes” on page 258.

You can view the offset from the current time for attributes that display a timestamp, such as Task Due D/T, by moving the mouse cursor on top of the attribute. The offset is displayed in a hint.

You can also hover your mouse over an task attribute and view the whole value of the task attribute as a tooltip.

Task History

Task History can be viewed by clicking the `History` tab in the Task Details view. It displays all of the task attributes' values and the task history events. Table 41 lists the attributes and actions that are available in the Task History.

Table 41: Task History Attributes

Attribute/Action	Event	Description
Date/Time		The date/time when the event occurred.
Actor		Shows who triggered the event. This is empty for iWD system-triggered events.
Event Code	`NEW`	Task has just been created.
	`CLASSIFY_START`	Task classification has started.
	`CLASSIFY`	Task classification has finished.
	`PRIORITIZE_START`	Task initial prioritization has started.
	`PRIORITIZE`	Task initial prioritization has finished.
	`HOLD`	Task is held. This can be triggered by the task source system or by the user with the task Held operation (see page 255).

Table 41: Task History Attributes (Continued)

Attribute/Action	Event	Description
Event Code (cont)	`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 255).
	`REJECT`	Task was rejected. Either the department or the process to which the task was associated is expired or not yet active.
	`ERROR`	A processing error has occurred, and the task was held for that reason.
	`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.
	`UPDATE_COMPLETE`	Task attributes are updated after a task is completed.
	`ASSIGN`	Task is assigned to an agent.
	`FINISH`	Agent has finished working on the task.
	`FINISH_RETURN`	Agent has returned the task to queue.

Table 41: Task History Attributes (Continued)

Attribute/Action	Event	Description
Event Code (cont)	`STOPPED`	That task has stopped (removed from database).
	`COMPLETE`	Task is completed
	`QUEUE`	The task is queued.
	`DISTRIBUTE_QUEUE`	The task is put into a queue.
	`DISTRIBUTE_WORKBIN`	The task is put into a workbin.
	`REPRIORITIZE`	Task reprioritization has finished.
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 249.

Hold Tasks

The Hold Tasks operation holds the selected task. When a task is held, it will not be reprioritized or distributed, but it can be canceled, updated, restarted or resumed.

Resume Tasks

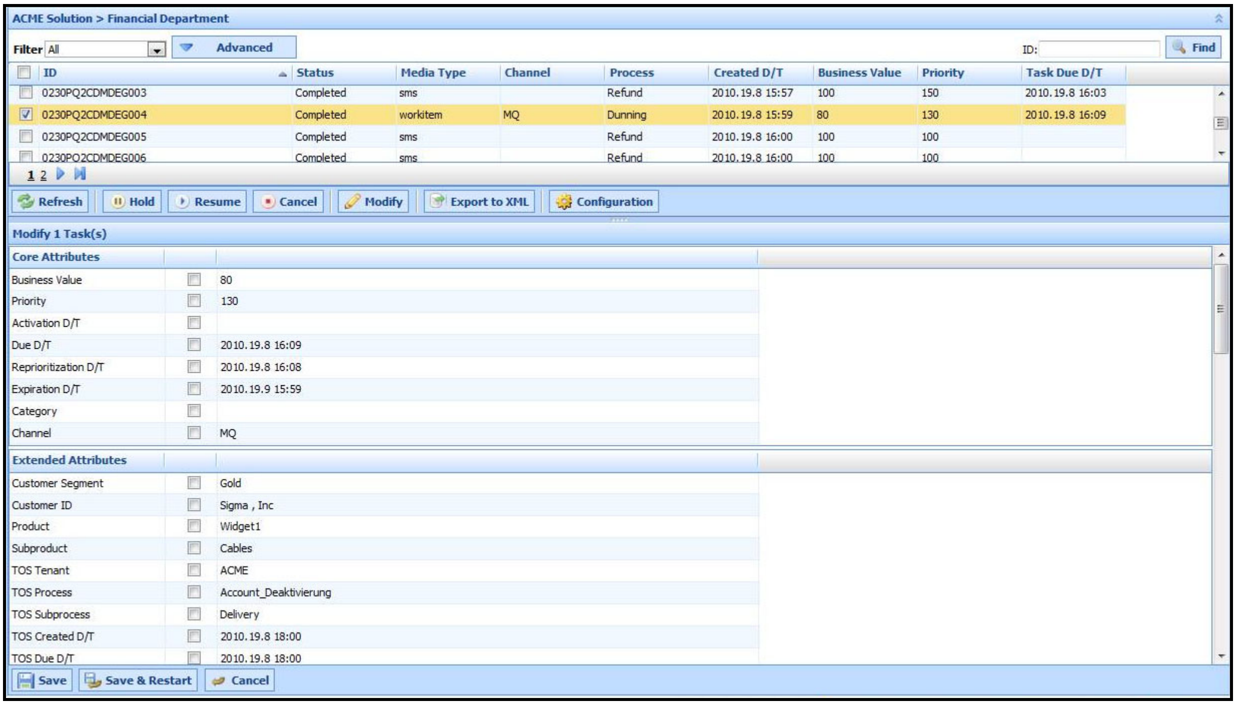
The Resume Tasks operation resumes processing of a held task. Only held tasks can be resumed.

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.

Modify Tasks

The Modify Tasks operation allows an update of a number of task attributes and, optionally a restart of the task. Figure 75 shows an example.



☐	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	
☐	0230PQ2CDMDEG006	Completed	sms		Refund	2010.19.8 16:00	100	100	

Modify 1 Task(s)

Core Attributes		
Business Value	☐	80
Priority	☐	130
Activation D/T	☐	
Due D/T	☐	2010.19.8 16:09
Reprioritization D/T	☐	2010.19.8 16:08
Expiration D/T	☐	2010.19.9 15:59
Category	☐	
Channel	☐	MQ

Extended Attributes		
Customer Segment	☐	Gold
Customer ID	☐	Sigma , Inc
Product	☐	Widget1
Subproduct	☐	Cables
TOS Tenant	☐	ACME
TOS Process	☐	Account_Deaktivierung
TOS Subprocess	☐	Delivery
TOS Created D/T	☐	2010.19.8 18:00
TOS Due D/T	☐	2010.19.8 18:00

Figure 75: 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.

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 8.1 Reference Manual* for more information about this function.

Note: The `setUpdateTrigger` function was introduced in URS 8.0.1.

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.

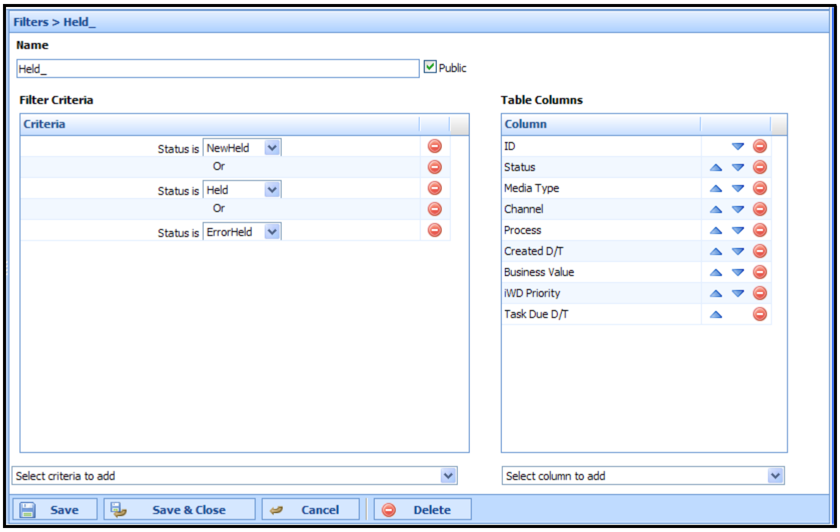
Export Tasks to XML

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.

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 76 shows an example.



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

Figure 76: Filters

Figure 42 lists the attributes and actions that are available in Filters view.

Table 42: 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). See "Security Policies" on page 146.
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 `Select criteria to add` drop-down list. Some criteria conditions are parameterized; for such conditions, parameters can be configured directly in a criteria table (such as, status for `Status is ...` criteria). Custom attributes can be used in many of the filter criteria, with proper configuration. See "Configuring Custom Attributes" on page 258.
Table Columns	The Global Task List will display these columns when the filter is selected. Columns can be added, removed, and reordered. A user can select from a list that includes all iWD core and extended attributes, by default. With proper configuration, custom attributes can be also be selected. See "Configuring Custom Attributes" on page 258.
Save, Save & Close, Cancel, Delete	Standard iWD Manager functions as described in "iWD Manager Overview" on page 127.

Configuring Custom Attributes

When you capture a task from a source system and that task has custom attributes in it, you will need to configure the system properly so that each custom attribute is recognized. Several steps are required to ensure the custom attribute:

- Appears on the Global Task List with a user-friendly label and can be used in advanced and custom filters.
- Can be properly populated in iWD Data Mart.
- Can be used in the `Condition`, `Order`, and `Segmentation` tabs of `Views` in Genesys Business Processes.

To configure the custom attribute complete the tasks in the Task Summary: Configuring a Custom Attribute.

Task Summary: Configuring a Custom Attribute

Objective	Related Procedures and Actions
Add database columns and Interaction Custom Property for Custom Attributes.	1. Add a unique column in the Interaction Server database interactions table to store the value of this custom attribute.
	2. Create a new Business Attribute to correspond to the custom attribute and map it to the new database column you added in the Interaction Server database interactions table.
	3. Add a unique column to two tables in the Interaction Server Event Log database to store the value of this custom attribute.
	4. Add two new options on the Interaction Server Event Log Database Access Point application to refer to the newly-added database columns in the Interaction Server Event Log database. To actions 1 to 4 in this task summary, see the Procedure: Adding new database column(s) and Interaction Custom Property for each Custom Attribute, on page 259.
Edit the resource file on the application server.	5. Edit this file to indicate how you want the custom attribute to be labeled on the Global Task List. See Procedure: Editing the iWD Manager's Resource File, on page 261

Procedure: Adding new database column(s) and Interaction Custom Property for each Custom Attribute

Purpose: To add new database columns and interaction custom properties to custom attributes.

Start of procedure

1. Decide on the name of the interaction user data key that will store the value of your custom attribute.

   This is the value you will use in the `CreateTask` message when you create new task from an iWD capture point (see description of the `CreateTask` message on page 295).

Note: If your custom attribute is going to be used to store a timestamp, ensure the name of the interaction user data key ends with `DateTime,` (for example, `CustomDateTime`). Then, the custom attributes that appear on the `Attributes` tab of the Global Task List will be properly formatted as date and time. For example, `December 31, 2012 9:30 PM,` instead of `2012-12-31T21:30:00Z.`

2. In your database server's Administration Console, add a new column to the Interaction Server database interactions table.

   This column can be a timestamp, string, or number. The exact data types will differ depending on the type of database server you are using. The name of this column does not necessarily need to match the interaction user data key that is storing the custom attribute.

Notes: If your custom attribute is going to be used to store a timestamp, ensure this column name ends with `DateTime,` (for example, `CustomDateTime`). Then, when you create a Global Task List filter with this custom attribute or use it in an advanced filter, the user will see a calendar control to pick the date,

Properties with the `Timestamp` data type cannot be used on the `Segmentation` tab of `Views` in a Genesys Business Process.

If you specify a custom database 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, which will be rejected by the DBMS.

3. In Genesys Administrator or Configuration Manager, create a new Business Attribute under the tenant you are working with for this iWD Solution (if an attribute already exists go to the next step):

   a. Create a new Business Attribute with the following properties:

      Name: `InteractionCustomProperties`

      Display Name: `Interaction Custom Properties`

      Type: `Custom`

   b. Expand `Interaction Custom Properties` and open `Attribute values.`

      The name of the `Attribute value` must match exactly, the interaction user data key name that you used in Step 1. The matching of names is case-sensitive. (You can create a separate display name.)

   c. In the new `Attribute value,` go to the `Annex` tab and create a section named `translation.`

d. In the translation section, create an option named translate-to, with a value name that matches the name of the database field you created in Step 2.

Note: Steps 4 and 5 will affect the correct population of the custom task attribute in iWD Data Mart and ensure that the Global Task List's History tab is correctly populated. If you complete these steps, the History tab will display an event whenever the value of the custom attribute is updated.

4. In your database server's Administration Console, add a new column to the Interaction Server Event Log database, in both the rpt_esp and rpt_interaction tables.

   Use the same data type for this column as you did in Step 2.

5. Create two new options on the Interaction Server Event Log Database Access Point application, as follows:

   a. Using Genesys Administrator or Configuration Manager, open the Interaction Server Event Log Database Access Point application object.

   b. On the Options tab, in the esp-custom-data section, add a new option with a name that matches the attached data key from Step 1, and a value that matches the new database column that you added in Step 4.

   c. In the itx-custom-data section, add a new option with a name that matches the attached data key from Step 1, and a value that matches the new database column added in Step 4.

6. Restart Interaction Server.

End of procedure

Editing the iWD Manager’s Resource File

By default, the on-screen labels you will see on the Global Task List's Attributes tab of the Task Details panel for all custom task attributes will be the interaction user data keys. These labels may not be very user-friendly. For example they might include multiple words concatenated, or they might use underscore characters or have odd capitalization.

Procedure: Editing the iWD Manager’s Resource File

Purpose: To configure user-friendly labels for custom task attributes by adding into a resources file, the labels that you want to see and that will be used by iWD Manager.

Start of procedure

1. Stop your application server.
2. Inside the `iwd_manager` application folder, which will be in the `webapps` folder of your application server, navigate to the `.../WEB-INF/classes/evo/cmc/ui/resources` directory, or if you are using a localized version of iWD Manager, navigate to the `.../WEB-INF/classes/lang/evo/cmc/ui/resources` directory.
3. Create a new text file called `resources_custom.properties` (if it does not already exist).
4. In the `resources_custom.properties` file, add a line for each custom attribute for which you want to define an on-screen label, by using the following format:

   `TASK_attribute_key=Attribute Label`

   Where:

   — `attribute_key` matches the interaction user data key of the custom attribute from Step 1 in Procedure: Adding new database column(s) and Interaction Custom Property for each Custom Attribute, on page 259.

   — `Attribute Label` is the user-friendly string you want to display on the Global Task List. For example, if your custom attribute's user data key is `MyAttribute`, but you want it to display as `My Custom Attribute` on the Global Task List, your entry would look like this: `TASK_MyAttribute=My Custom Attribute`.

   Note: This label will be used in the following places in the Global Task List:

   - As the label for this attribute on the `Attributes` tab of the `Task Details` panel.
   - In the `Select` columns to add drop-down list on the `Filters` page.
   - As the column header if you add this custom attribute as a column in the task table when you are creating a custom filter.

5. In the same `resources_custom.properties` file that was used in the Procedure: Adding new database column(s) and Interaction Custom Property for each Custom Attribute, on page 259, add a new line for each custom attribute, by using the following format:

   `FILTER_ATTR_attributedb_column_name=Attribute Label`

   Where:

   — `attributedb_column_name` matches the column name that you added to the interactions table in Step 2 in the Procedure: Adding new database column(s) and Interaction Custom Property for each Custom Attribute, on page 259.

— `Attribute Label` is the user-friendly string you want to display in the Global Task List. For example, if the name of the database column is `my_attribute`, this line in the `resources_custom.properties` file would look like this: `FILTER_ATTR_my_attribute=My Custom Attribute`.

Note: This label will be used in the following places in the Global Task List:

- In the `Advanced Filters` drop-down list.
- In the `attributes` drop-down list that you use to add a custom attribute in a filter criterion, when you are creating or editing a Global Task List filter.

6. Restart the `iwd_manager` application on your application server for the changes to become effective.

Note: If the user-friendly labels need to be localized, create a separate file for each supported locale. Use the following name pattern for the file: `resources_custom_xx[_YY].properties`, where `xx` and `YY` are language code and country code, respectively. Also, do not use Unicode or any language-specific encoding for the properties files. Use ISO-8859-1 encoding, which supports only a few Western languages. Encode resources in other languages by using escape sequences, such as the native2ascii program, which is a part of the Oracle JDK.

End of procedure

Chapter

8 High Availability

This chapter describes high-availability and redundancy in iWD. This chapter is divided into the following sections:

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 77 illustrates this configuration.

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.

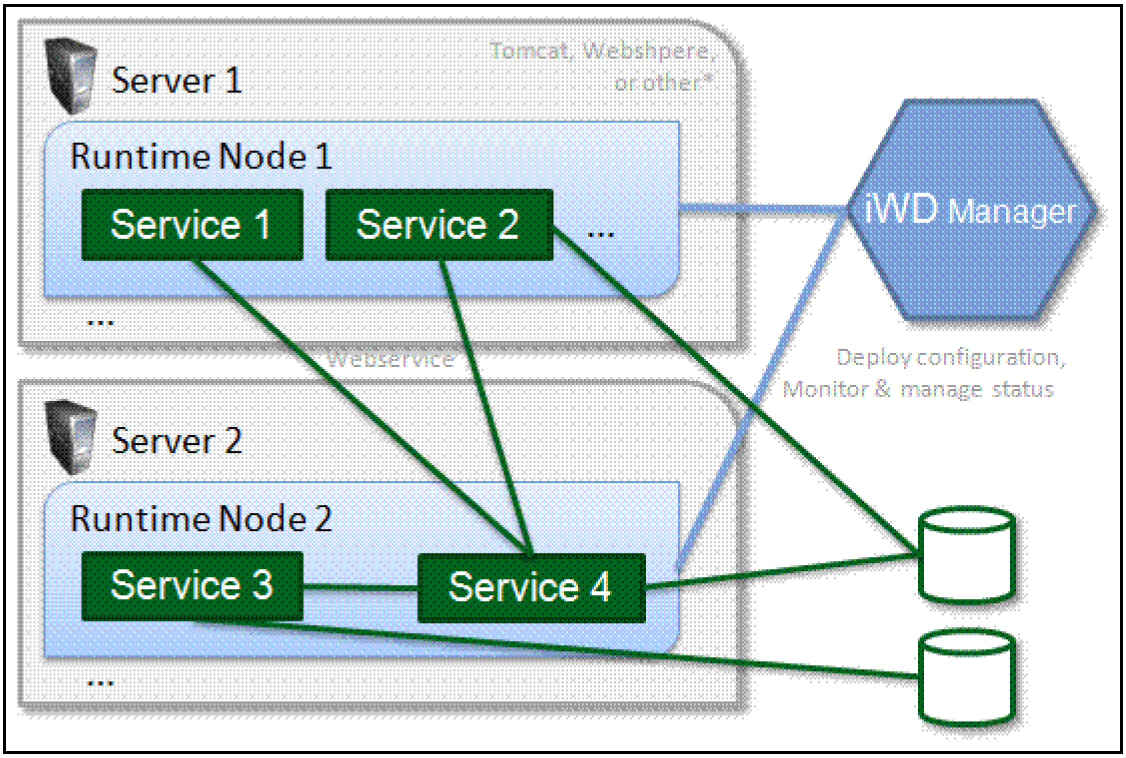


Figure 77: Distributed Deployment with Centralized Management

High Availability for iWD Components

In iWD 8.1, the following legacy iWD Capture Points support hot standby: XML File Capture Point, Database Capture Point, and WebSphere MQ Capture Point.

The Interaction Server Integrated Capture Points support warm standby. For more information about this support, see the *eServices 8.1 Deployment Guide*.

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 prepare the interaction user data prior to the Genesys Rules Engine being invoked from a routing strategy. 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 can 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 187 and “Configuration of List Objects” on page 314.

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 DVD 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 314) 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) 8.1 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 External Service Protocol (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 iWD Data Mart, specifying the iWD Runtime Node to which it is being deployed.

End of procedure

Next Steps

- Configure iWD Services in iWD Manager, specifying the desired Runtime Node.

Interaction Server Redundancy

For information about high-availability and how Interaction Server supports redundancy, see the *eServices 8.1 User's Guide*.

Appendix

A Task Attributes and Interaction Properties

This appendix describes the task and attribute properties that are supported in iWD 8.1.

It contains the following sections:

Task Attributes

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 8.1 User's Guide*.

Table 43 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.

For details about how these task attributes are used in iWD messages such as CreateTask, see "iWD Messages" on page 291 in Appendix B.

Table 43: 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)

Table 43: Task Attribute Mapping (Continued)

Task Attribute	Column Name in Interactions Table	Type	Interaction Attached Data Key	Label in Global Task List
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	Integer	IWD_businessValue	Business Value
DueDateTime	IWD_dueDateTime	Timestamp	IWD_dueDateTime	Task Due D/T
Priority	priority	Integer	Priority	Priority
ReprioritizeDateTime	IWD_reprioritizeDateTime	Timestamp	IWD_reprioritizeDateTime	Reprioritization D/T
AssignedToUser	assigned_to	String	RTargetAgentSelected	Assigned To
AssignedDateTime	assigned_at	Timestamp	AssignedAt	Assigned D/T
-	completed_at	Timestamp	CompletedAt	Completed D/T
ExpirationDateTime	IWD_expirationDateTime	Timestamp	IWD_expirationDateTime	Expiration D/T
-	IWD_solutionId	String	IWD_solutionId	-
Extended Attributes				
CustomerSegment	IWD_ext_customerSegment	String	IWD_ext_customerSegment	Customer Segment

Table 43: Task Attribute Mapping (Continued)

Task Attribute	Column Name in Interactions Table	Type	Interaction Attached Data Key	Label in Global Task List
CustomerId	IWD_ext_customerId	String	IWD_ext_customerId	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	String	IWD_ext_ requestedAgent Group	Requested Agent Group
RequestedPlace Group	IWD_ext_requestedPlace Group	String	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
SourceFirst CreatedDateTime	IWD_ext_sourceFirst CreatedDT	Timestamp	IWD_ext_source FirstCreatedDate Time	-
SourceCreatedDate Time	IWD_ext_sourceCreated DateTime	Timestamp	IWD_ext_source CreatedDateTime	TOS Created D/T
SourceDueDate Time	IWD_ext_sourceDueDateTime	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

Table 43: Task Attribute Mapping (Continued)

Task Attribute	Column Name in Interactions Table	Type	Interaction Attached Data Key	Label in Global Task List
Archive Destination[b]	in <flexible_properties>	String	Archive Destination	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.

[b] Archive Destination has been maintained from iWD 8.0 and 8.1.0 releases. However, due to changes in task archiving, it is no longer necessary to populate this attribute through business rules. See "Task Archiving" on page 21.

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.
8. If required, configure user-friendly labels for any custom attribute, that will appear in the Global Task List's `Attributes` tab, in the list of `Advanced Filters,` and the list of attributes that are used when you are building custom filters. See "Configuring Custom Attributes" on page 258.

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.

Appendix

B Legacy iWD Capture Point Services

This chapter contains information about the legacy iWD Capture Point services. These Capture Point services were first provided in iWD 7.6.1 and were carried forward to iWD 8.0.

If you are deploying a new iWD 8.1 Solution (as opposed to upgrading to iWD 8.1 from 7.6.1 or 8.0), Genesys recommends you use the Capture Points that are integrated with Interaction Server (also known as *Integrated Capture Points,* or *ICPs*). The information in this chapter is provided for those customers who choose to continue to use the legacy Capture Point services.

It contains the following sections:

Overview

Capture points represent an interface that feeds tasks to iWD. There are three out-of-the-box legacy iWD Capture Point services:

- Web Service Capture Point Service—Enables third-party applications to submit tasks to iWD through a Web service interface. It also provides an interface that is used to manage submitted tasks, such as holding and restarting.
- Database Capture Point Service—Captures tasks directly from database tables. Configuration of this Capture Point Service is based on Structured Query Language (SQL).

- XML File Capture Point Service—Captures tasks from Extensible Markup Language (XML) files. The files must comply with the iWD Task XML schema.

The XML File Capture Point Service supports optional message transformation, where a transformation script is used to transform the inbound message from the native source system XML format to the native iWD XML format. Likewise, transformation of outbound messages from iWD to the source system is supported. These scripts are written in the Groovy scripting language and are executed by the iWD Scripting Service.

Important Information About Legacy iWD WebSphere MQ Capture Point Service

In iWD 7.6.1 and 8.0, Genesys also offered an optional WebSphere MQ Capture Adapter. (In iWD 7.6.1, the product was known as the iWD MQ Series Capture Adapter.) In iWD 8.1, this optional Capture Adapter (Capture Point) is no longer offered. Instead, Genesys recommends that iWD 8.1 customers, who need to connect to a source system through the IBM WebSphere MQ Server, do so through the iWD JMS Integrated Capture Point (which is sold under the name *v8.1 - iWD JMS Capture Adapter*). If you have an active support agreement and previously purchased the iWD MQ Series Capture Adapter or iWD WebSphere MQ Capture Adapter, you are entitled to use the iWD JMS Capture Adapter license at no cost. For more information about this license, contact your Genesys sales representative.

Installing and Configuring Legacy iWD Capture Points

This section describes the installation and configuration of the legacy Capture Points (Web Service, XML File, and Database) and also describes the configuration properties for these services.

To use one of the legacy iWD Capture Point services, the services must be created and deployed under an iWD Solution, on a Runtime Node. The first step is to import the iWD Capture Point service templates so that they can be used to create new services.

Note: Importing the template is not necessary if you are upgrading from a previous iWD release to iWD 8.1and have already imported the capture point services to your iWD 8.0 deployment. When the iWD configuration database is upgraded to 8.1 (which occurs automatically the first time you launch iWD Manager 8.1), the Capture Point services will still be present in the configuration.

Before importing the service templates, determine if you will need to use XML message transformation. This will be used by the legacy XML File Capture Point Service, if the inbound XML messages are not already in the iWD XML message format. In that case, the message can be transformed by a Groovy script that is executed by the iWD Scripting Service. Sample inbound and outbound Groovy transformation scripts, along with the service template for the Scripting Service, are included in the `iwd_transform.xml` configuration file.

Procedure: Importing the Capture Point and Transformation configuration XML files to iWD Manager

Purpose: To import the Capture Point configuration XML file and Transformation configuration XML file (optional) to iWD Manager

Summary

In this procedure, importing the Transformation configuration XML file is optional.

Prerequisites

- iWD Manager, the iWD Runtime Node, and all required components are installed.
- The Application servers are started. (If applications are running on WebSphere, they are also started.)

Start of procedure

1. Launch and log in to iWD Manager.
2. In the `General` section of the `System` tenant, open `Import/Export.`
3. From the `<iWD Manager installation directory>\config` directory, import the `iwd_capture.xml` file to iWD Manager.
4. (Optional) If you plan to use XML message transformation, from the `<iWD Manager installation directory>\config` directory, import the `iwd_transform.xml.`

   A message will appear on the bottom of the screen after each import, to indicate that the import was successful.

End of procedure

Next Steps

- Create the Capture Point service(s) that you need, under your iWD Solution. See Procedure: Creating a new Capture Point service under your iWD Solution, on page 280.
- Optionally, if you are using message transformation, create a new Scripting Service under your Solution.

Procedure: Creating a new Capture Point service under your iWD Solution

Purpose: To create a new Capture Point service under the iWD Solution.

Start of procedure

1. Launch and log in to iWD Manager.
2. Open the `iWD Managed Tenant` under which your Solution is configured.
3. On the navigation bar, open `Services`.
4. First expand your `iWD Solution` and then, the `Services` node.
5. Click the `New Service` link.
6. Select the appropriate service template from the drop-down list for the type of legacy Capture Point service you want to configure.
7. In the `Description` field, enter a description.

   Note: A description is not required, but can be populated, if desired.

8. Enter the configuration properties for the service.

   For a description of all of the configuration properties for each of the iWD legacy Capture Point services, see "Legacy Capture Point Service Details" on page 281.
9. Save the service.
10. Deploy the Solution.

End of procedure

Next Steps

- If you are using XML message transformation, repeat the steps in this procedure to add a Scripting Service to your Solution, making the relevant changes where necessary.

- Create inbound and outbound Groovy transformation scripts for use in your Scripting Service. Sample transformation scripts are included in the iwd_transform.xml configuration file.

Legacy Capture Point Service Details

This section provides details about the iWD legacy Capture Point Service.

Web Service Capture Point

The Web Service Capture Point Service 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 171, the properties listed in Table 44 are configurable for the Web Service Capture Point.

Table 44: 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 Details" on page 171 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 183 for descriptions of these common properties.
checkIfAlreadyCaptured	If true, iWD will verify whether another task that has a given captureId has already been captured.

Table 44: Web Service Capture Point Properties (Continued)

Property	Description
`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 185) 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/iwd_node/services/webserviceCapturePoint`. To retrieve a WSDL file for the Webservice, attach `?WSDL` to the URL.
`stringOnlyParameters`	If selected, only string parameters will be accepted.

`timeZoneIsSupplied`	**If this property is selected** (`timezoneIsSupplied` = `ON`), then:
	• 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.
	The time format includes the timezone designator and the time without milliseconds, separated by a `T`:
	`(yyyy-MM-dd'T'HH:mm:ssZZ).`
	For example:
	`2008-07-21T21:00:00+01:00`
	If this property is not selected (`timezoneIsSupplied` = `OFF`), then:
	• 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 291) 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 171, the properties listed in Table 45 are configurable for the XML File Capture Point.

Table 45: 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 `Default` unless otherwise instructed by Genesys Technical Support. See "Service Log Levels" on page 182 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 183 for descriptions of these common properties.
checkIfAlreadyCaptured	If true, iWD will verify whether another task that has a given `captureId` 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.

Table 45: XML File Capture Point Properties (Continued)

Properties	Description
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 185) 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 `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 291). If specified, XML input/output will be transformed using input/output transformation scripts.

Table 45: XML File Capture Point Properties (Continued)

Properties	Description
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 171, the properties listed in Table 46 are configurable for the Database Capture Point.

Table 46: 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 Default unless otherwise instructed by Genesys Technical Support. See "Service Log Levels" on page 182 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 183 for descriptions of these common properties.
checkIfAlreadyCaptured	If true, iWD will verify whether another task that has a given captureId 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.
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 core iWD attribute mediaType may be specified in the CreateTask message. If the mediaType attribute is returned as a result of the query configured in captureQuerySQL, then it will override the default mediaType specified in the Database Capture Point service property defaultMediaType.
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 185) 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.

Table 46: Database Capture Point Properties (Continued)

Property	Description
`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 `emergencySleepThreshold` 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 `captureQuerySql` parameter. The value of the field is mapped to the `captureId` 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.
`fieldsToAttach`	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. For example: `field1,field2,special*` This expression means that the fields that have the names `field1` and `field2` will be attached, as well as any field that has a name that starts with `special`.
`captureQuerySql`	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 `idField` parameter. For example: `select * from TABLE where STATUS="new"`
`capturedUpdateSql`	The database update statement that updates the database to reflect that certain data has already been captured as a task in iWD. The `captureId` parameter can be used to reference the particular row. For example: `update TABLE set STATUS='captured' where idField=:captureId`

Table 46: Database Capture Point Properties (Continued)

Property	Description
`errorUpdateSql`	The database update statement that updates the database to reflect that the associated task in iWD has been held by an error. The `captureId` parameter can be used to reference the particular row. For example: `update TABLE set STATUS='error' where idField=:captureId`
`assignedUpdateSql`	The database update statement that updates the database to reflect that the associated task in iWD has been assigned. The `captureId` parameter can be used to reference the particular row. The `userId` parameter can be used to set the name of the user who is working on the task. For example: `update TABLE set STATUS='assigned', USER=:userId where idField=:captureId`
`completedUpdateSql`	The database update statement that updates the database to reflect that the associated task in iWD has been completed. The `captureId` parameter can be used to reference the particular row. For example: `update TABLE set STATUS='completed' where idField=:captureId`
`canceledUpdateSql`	The database update statement that updates the source database to reflect that the associated task in iWD has been canceled. The `captureId` parameter can be used to reference the particular row. For example: `update TABLE set STATUS='canceled' where idField=:captureId`
`heldUpdateSql`	The database update statement that updates the source database to reflect that the associated task in iWD has been put on hold. The `captureId` parameter can be used to reference the particular row. For example: `update TABLE set STATUS='held' where idField=:captureId`
`queuedUpdateSql`	The database update statement that will be executed when a task, captured by the Database Capture Point, is queued. For example: `update TABLE set STATUS='queued' where idField=:captureId`

Table 46: Database Capture Point Properties (Continued)

Property	Description
errorHeldUpdateSql	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 captureId parameter can be used to reference the particular row. For example: update TABLE set STATUS='errorHeld' where idField=:captureId
rejectedUpdateSql	The database update statement that will be executed when a task, captured by the Database Capture Point, is rejected. For example: update TABLE set STATUS='rejected' where idField=:captureId
restartedUpdateSql	The database update statement that updates the database to reflect that the associated task in iWD has been restarted in iWD. The captureId parameter can be used to reference the particular row. For example: update TABLE set STATUS='restarted' where idField=:captureId
updatedUpdateSql	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. For example: update UPDATE_TABLE set PRIORITY=:priority where idField=:captureId
sourceUpdateQuerySql	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. For example: 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

Table 46: Database Capture Point Properties (Continued)

Property	Description
`sourceUpdatedUpdateSql`	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. For example: `update UPDATE_TABLE set processed='T' where idField=:captureId`
`sourceErrorUpdateSql`	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`.
`captureBatchSize`	Performance tuning: 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.
`sourceUpdateBatchSize`	Performance tuning: 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.
`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: 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.

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 283) and WebSphere MQ Capture Point (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 8.1 User's Guide* for more information.

In addition to the options and actions described in "Service Details" on page 171, the properties listed in Table 47 are configurable for the Scripting Service.

Table 47: 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 182 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 183 for descriptions of these common properties.

iWD Messages

This section describes the native XML message format that is supported for the Legacy iWD Capture Point services, as well as for some of the Integrated Capture Points. For example, the JMS and XML File Integrated Capture Points support these native iWD XML messages when working in `iWD mode`, which uses Groovy transformation scripts to transform the source system's XML message format into XML messages that adhere to the message schema defined in this section.

Note: Database Capture Point does not support native XML. Webservice Capture Point uses its own SOAP-compatible format.

In addition, a detailed description is provided for all of the input and output iWD Messages that are supported, including the following information 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 48 on page 292) 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 271, 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 48 describes the data types used in iWD messages.

Table 48: Data Types for iWD Messages

Type	Description
Integer	An integer value (-2^{31} <value <2^{31}).
String	A string value. The maximum length is specified in parentheses, where applicable.
Boolean	A Boolean value (`true` or `false`).
DateTime	A date/time value. Date/time should be formatted according to the ISO 8601 standard `YYYY-MM-DDThh:mm:ss` (for example 2007-08-26T21:32:00) and should be provided for the time zone that is configured for the given 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 49.

Table 49: 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 297).

Error codes: See Table 50.

Table 50: 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 295).

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 51.

Table 51: 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 mandatory. However, the value can be null.

Format:

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

Standard action attributes, as documented in “Task Action” on page 293, 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 52.

Table 52: Attributes for CreateTask Messages

Attribute	Description
Hold	Whether to hold the task initially. If true, the task will be created with its initial status set to `NewHeld` and will not be processed further, until a subsequent `ResumeTask` message (see "ResumeTask" on page 308).
CaptureId (optional)	If a `CaptureId` is not provided, it will be assigned to the same generated value as `BrokerId`.
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 293 and "TaskInfo" on page 297 for the description of the remaining attributes.
Response message: `TaskCreated` (see "TaskCreated" on page 296).
Error codes: see Table 53.

Table 53: 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 295) and indicates successful task creation.
Format:

```
<GTLMessages>
   <TaskCreated>
```

Standard notification attributes, as documented in "Task Notification" on page 294.

```
   </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 54.

Table 54: 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 293.
   </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 297) and provides detailed information about the requested task.

Format:

```
<GTLMessages>
   <TaskInfo>
      Standard action attributes, as documented in "Task Action" on
page 293.
      <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 55.

Table 55: 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.
DepartmentId	The tasks's department ID, as configured in iWD Manager, assigned when the task's process is identified either by iWD rules or explicitly by the task-originating system. This attribute is submitted to the CIM Platform with the `IWD_DepartmentId` key; updates in the CIM Platform are ignored.
processId	The tasks's process ID, as configured in iWD Manager, assigned when the task's process is identified either by iWD rules or explicitly by the task-originating system. This attribute is submitted to the CIM Platform with the `IWD_processId` key; updates 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 `IWD_channel` 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 `IWD_category` key; updates in the CIM Platform are picked up.

Table 55: Attributes for TaskInfo Messages (Continued)

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

Table 55: Attributes for TaskInfo Messages (Continued)

Attribute	Description
priority	The task priority, which is an integer value that is used to order tasks. The higher the value, the higher 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`

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.

Table 55: Attributes for TaskInfo Messages (Continued)

Attribute	Description
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_sourceFirstCreatedDateTimeTime` key; updates in the CIM Platform are ignored.
sourceCreatedDateTime	The task-creation timestamp in the task-originating system. This attribute is submitted to the CIM Platform with the `IWD_ext_sourceCreatedDateTime` key; updates in the CIM Platform are ignored.
sourceDueDateTime	The task-due timestamp in the task-originating system. This attribute is submitted to the CIM Platform with the `IWD_ext_sourceDueDateTime` key; updates in the CIM Platform are ignored.
sourceProcessType	A related process in the task-originating system—for example: Order. This attribute is submitted to the CIM Platform with the `IWD_ext_sourceProcessType` key; updates in the CIM Platform are ignored.
sourceProcessTypeSubtype	The subtype of the related process in the task-originating system. This attribute is submitted to the CIM Platform with the `IWD_ext_sourceProcessSubtype` key; updates in the CIM Platform are ignored.
sourceTenant	The tenant ID or name in the task-originating system. This attribute is submitted to the CIM Platform with the `IWD_ext_sourceTenant` key; updates in the CIM Platform are ignored.
requestedAgent	The agent requested for the task.
requestedSkill	The skill requested for the task.
requestedAgentGroup	The Agent Group requested for the task.

Table 55: Attributes for TaskInfo Messages (Continued)

Attribute	Description
requestedPlaceGroup	The Place Group 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 293.
      <category>String(32)</category>
      <activationDateTime>DateTime</activationDateTime>
      <dueDateTime> DateTime </dueDateTime>
      <expirationDateTime> DateTime </expirationDateTime>
      <businessValue>Integer</businessValue>
      <priority>Integer</priority>
      <ext>
         <customerID>String(64)</customerID>
         <customerSegment> String(64)</customerSegment>
         <productType> String(64)</productType>
         <productSubtype> String(64)</productSubtype>
         <resultCode> String(64)</resultCode>

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

`Attributes:` See “Task Action” on page 293 and “TaskInfo” on page 297 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 8.1 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 303), 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 294.

```
      <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 293 and "TaskInfo" on page 297 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 294.

```
      <Queue>String(255)</Queue>
      <QueueType>String(16)</QueueType>
      <QueueTarget>String(255)</QueueTarget>
   </TaskDistributedQueue>
</GTLMessages>
```

Attributes: See "Task Notification" on page 294 and "TaskInfo" on page 297 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 294.
        <AssignedToUser>String(64)</AssignedToUser>
    </TaskAssigned>
</GTLMessages>
```

Attributes: See “Task Notification” on page 294 and “TaskInfo” on page 297 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 293.
    </CompleteTask>
</GTLMessages>
```

Attributes: See “Task Action” on page 293 for a description of the attributes.

Response message: “TaskCompleted”.

Error codes: See Table 56.

Table 56: Error Codes for CompleteTask Messages

Error code	Description
CANNOT_COMPLETE_TASK	Cannot complete the task, because it is already completed.

TaskCompleted

Direction: Out

Description: The `TaskCompleted` message is submitted as a response to the `CompleteTask` message (see “CompleteTask” on page 306), 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 294
    </TaskCompleted>
</GTLMessages>
```

Attributes: See “Task Notification” on page 294 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 308).

Only tasks that are not held, completed, canceled, or rejected can be held.

Format:

```
<GTLMessages>
   <HoldTask>
      Standard action attributes, as documented in "Task Action" on page 293.
   </HoldTask>
</GTLMessages>
```

Attributes: See "Task Action" on page 293 for a description of the attributes.

Response message: "TaskHeld".

Error codes: See Table 57.

Table 57: Error Codes for HoldTask Messages

Error code	Description
CANNOT_HOLD_ASSIGNED_TASK	Cannot hold the task because it is 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 306), 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 294.
   </TaskHeld>
</GTLMessages>
```

Attributes: See "Task Notification" on page 294 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 294.
```

```
        <Error>String(255)</Error>
    </TaskErrorHeld>
</GTLMessages>
```

Attributes: See “Task Notification” on page 294 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 293.
    </ResumeTask>
</GTLMessages>
```

Attributes: See “Task Action” on page 293 for a description of the attributes.

Response message: “TaskResumed”.

Error codes: See Table 58.

Table 58: 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 308), as well as when a task is held from iWD Manager.

Format:

```
<GTLMessages>
    <TaskResumed>
        Standard notification attributes, as documented in “Task Notification” on
page 294.
    </TaskResumed>
</GTLMessages>
```

Attributes: See “Task Notification” on page 294 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 293.
  </RestartTask>
</GTLMessages>
```

Attributes: See “Task Action” on page 293 for a description of the attributes.

Response message: “TaskRestarted”.

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 294.
  </TaskRestarted>
</GTLMessages>
```

Attributes: See “Task Notification” on page 294 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 293.
  </CancelTask>
</GTLMessages>
```

Attributes: See “Task Action” on page 293 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. 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 294.
   </TaskCanceled>
</GTLMessages>
```

Attributes: See "Task Notification" on page 294 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 294.
   </TaskRejected>
</GTLMessages>
```

Attributes: See "Task Notification" on page 294 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 310), 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>
```

Appendix

C iWD Business Process (IWDBP)

This appendix describes the default iWD business process (IWDBP) that is supplied in the iWD Setup Utility component. It contains the following sections:

Software Requirements

The IWD Business Process that is described in this appendix requires Interaction Routing Designer 8.1.2 or higher.

- The *Universal Routing 8.1 Deployment Guide* describes how to have the Interaction Design shortcut bar appear in IRD, if it has not appeared automatically.
- The *Universal Routing 8.1 Business Process User's Guide* provides an in-depth discussion of business processes.
- The *Universal Routing 8.1 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.

Configuration of List Objects

The iWD Business Process (IWDBP) uses two Configuration Server List Objects.

The first List Object, Iwd_Esp_List, has three lists.

- The first is used to map the iWD Solution ID (IWD_SolutionId) to the name of the Business Context Management Service application configured in Configuration Server that will be used to invoke the Genesys Rules Engine.
- The second maps the iWD Solution ID to the name of the Genesys Rules Engine application.
- The third list (from release 8.1.1 onwards) maps the iWD Solution Runtime ID to the name of the Universal Contact Server (USCS) application. This is optional, and is used to allow the business logic in IWDBP to update the interaction record in the UCS database to mark the interaction as *done* (that is, the value of the Status column in the Interaction table will be set to 3) when it enters the iWD_Completed, iWD_Rejected, or iWD_Canceled queues.

The second List Object, Iwd_Package_List, maps the iWD Solution ID to the rules package that will be evaluated when the Genesys Rules Engine is invoked from the IWDBP business process.

Both of these List Objects must be correctly configured for IWDBP to work.

One business process can serve several solutions under the same tenant. The iWD Setup Utility automatically creates these two List Objects for the Solution you indicate in the Setup Utility. In environments with only one solution, no further configuration needs to be done on the List Objects. If you have multiple solutions (or add one at a later time) these two List Objects need to be updated.

Iwd_Esp_List

BCMSServiceList

In the `Iwd_Esp_List` List Object, the `BCMSServiceList` list looks like a list of pairs:

`Solution_1`	`ESPService_1`
`Solution_2`	`ESPService_2`
`Solution_i`	`ESPService_i`

Where the Solution ID is the key, and the name of the Business Context Management Service `Application` is the value.

GREServerList

The `GREServerList` list looks like a list of pairs:

`Solution_1`	`GREApplication_1`
`Solution_2`	`GREApplication_2`
`Solution_3`	`GREApplication_3`

Where the Solution ID is the key, and the name of the Genesys Rules Engine `Application` is the value.

ContactServerList

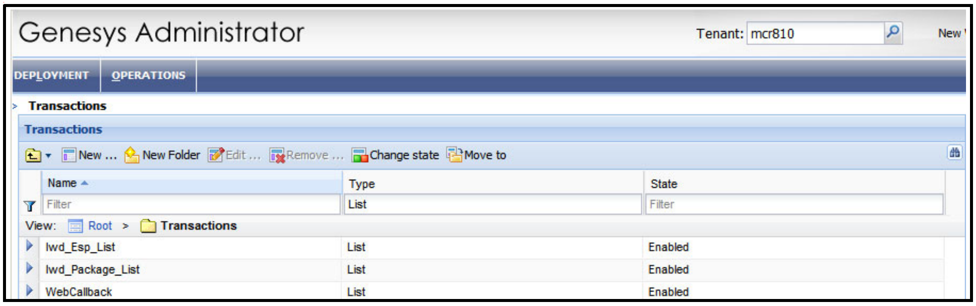
In release 8.1.1, an additional list, `ContactServerList` is included. The `ContacServerList` list looks like a list of pairs:

`iWD Solution Runtime_1`	`ContactServer_1`
`iWD Solution Runtime_2`	`ContactServer_2`
`iWD Solution Runtime_3`	`ContactServer_3`

Where iWD Solution Runtime ID is the key and the name of a Universal Contact Server associated with Interaction Server 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`. Similar issues will occur if the Genesys Rules Engine application or the Universal Contact Server application are incorrectly mapped.

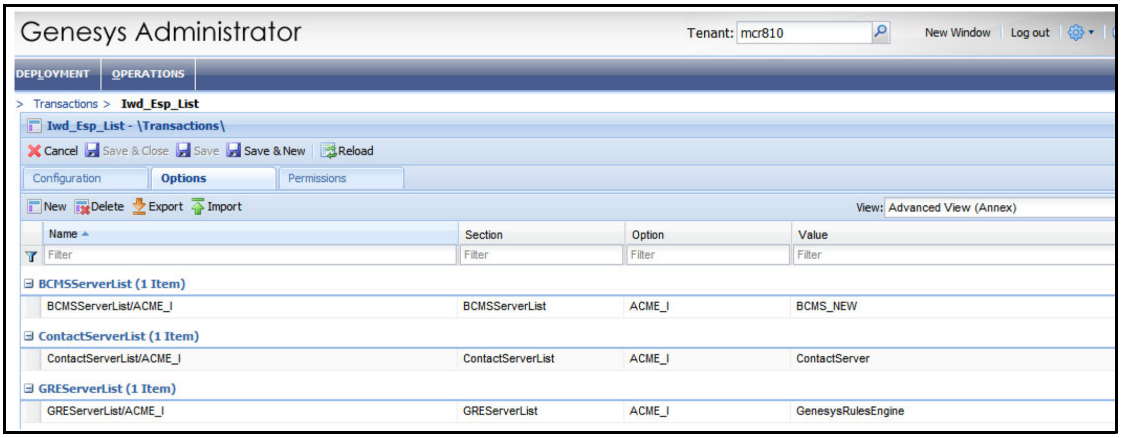
These key-value pairs in a List Object need to be set up only once per tenant, and can be configured in Interaction Routing Designer (IRD) or Genesys Administrator. See Figure 78



Name	Type	State
Filter	List	Filter
Iwd_Esp_List	List	Enabled
Iwd_Package_List	List	Enabled
WebCallback	List	Enabled

Figure 78: List Objects in IRD

Figure 79 shows the details of the List Object.



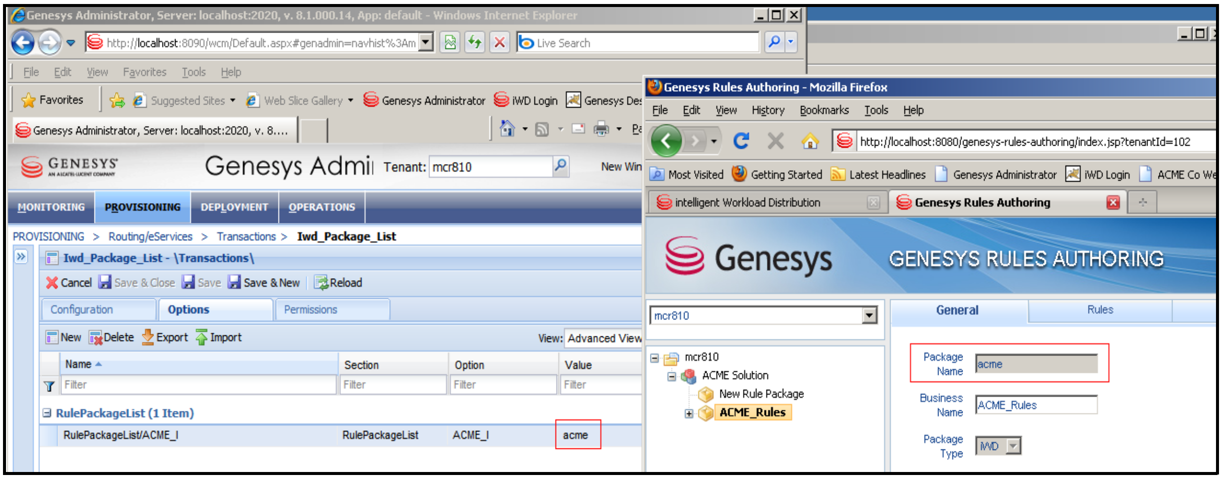
Name	Section	Option	Value
Filter	Filter	Filter	Filter
BCMSServerList (1 Item)			
BCMSServerList/ACME_I	BCMSServerList	ACME_I	BCMS_NEW
ContactServerList (1 Item)			
ContactServerList/ACME_I	ContactServerList	ACME_I	ContactServer
GREServerList (1 Item)			
GREServerList/ACME_I	GREServerList	ACME_I	GenesysRulesEngine

Figure 79: List Object Details

Iwd_Package_List

The Iwd_Package_List List Object is used to correlate the IWD Solution ID (IWD_SolutionId) to the name of the rule package that will be evaluated when requests are made to the Genesys Rules Engine from the IWDBP business process.

The Iwd_Package_List List Object contains a single list, RulePackageList. Create a new key/value pair for each iWD Solution that you have configured under your Configuration Server tenant, where the key or option is the IWD Solution ID and the value is the *Package Name* of the rules package. See Figure 80.


Genesys Administrator, Server: localhost:2020, v. 8.1.000.14, App: default - Windows Internet Explorer
http://localhost:8090/wcm/Default.aspx#genadmin=navhist%3Am
File Edit View Favorites Tools Help
MONITORING PROVISIONING DEPLOYMENT OPERATIONS
PROVISIONING > Routing/eServices > Transactions > Iwd_Package_List
Iwd_Package_List - \Transactions\
Cancel Save & Close Save Save & New Reload
Configuration Options Permissions
New Delete Export Import
View: Advanced View
Genesys Rules Authoring - Mozilla Firefox
http://localhost:8080/genesys-rules-authoring/index.jsp?tenantId=102
GENESYS RULES AUTHORING
General Rules
mcr810
ACME Solution
New Rule Package
ACME_Rules

Tenant: mcr810

Name	Section	Option	Value
RulePackageList (1 Item)			
RulePackageList/ACME_I	RulePackageList	ACME_I	acme

Package Name: acme

Business Name: ACME_Rules

Package Type: iWD

Figure 80: IWD_Package_List

iWD Business Process

The iWD business process (`IWDBP`) contains the following strategies:

- `Classification`
- `Prioritization`
- `Distribution`
- `Mark Interaction as Done`
- `Removal`

The iWD business process contains the following subroutines:

- `DetermineESPServerName`
- `DetermineRulePackageName`

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 display in the Global Task List with the status `Queued`.

Figure 81 shows the entire business process as it appears in the `Interaction Design` window of Interaction Routing Designer.

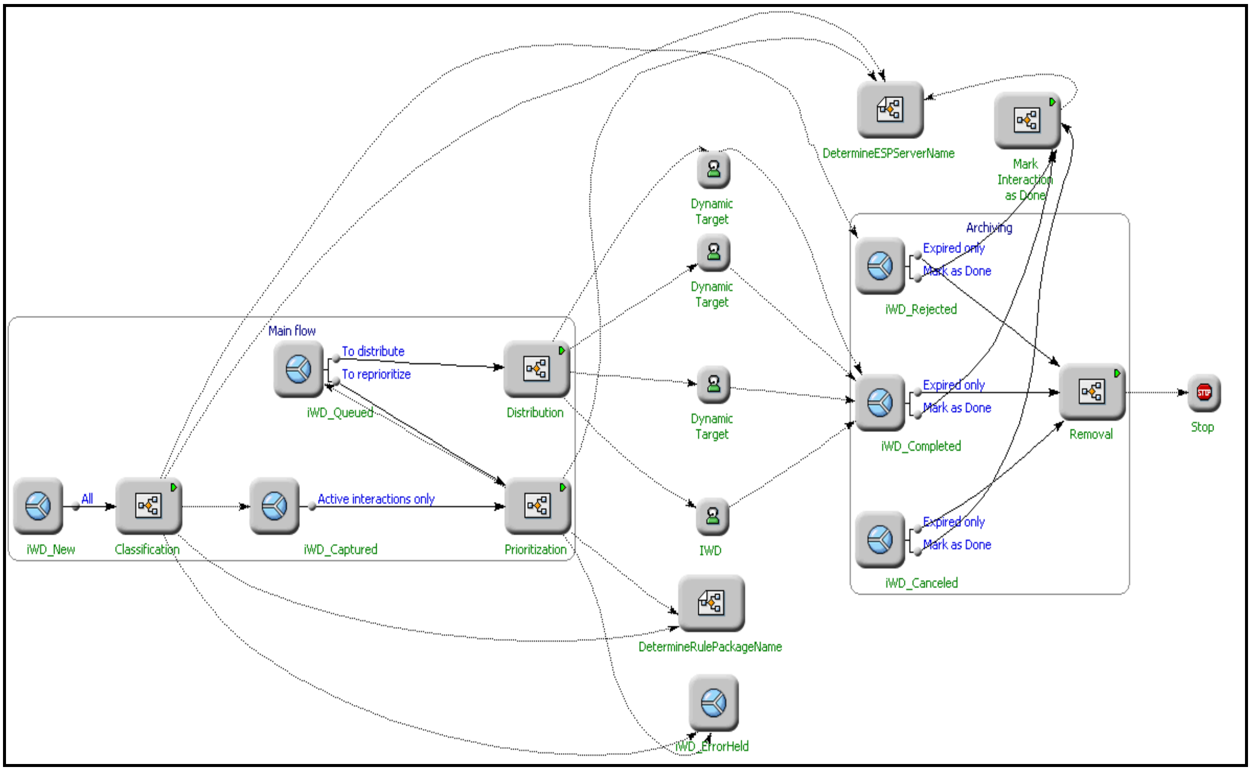


Figure 81: iWD Business Process

The group of objects on the left-hand side are part of the "Main Flow" of the business process. Figure 82 shows this section in more detail. The group of objects on the right-hand side represent the “Archiving” section of the business process. Figure 83 shows the Archiving section in more detail.

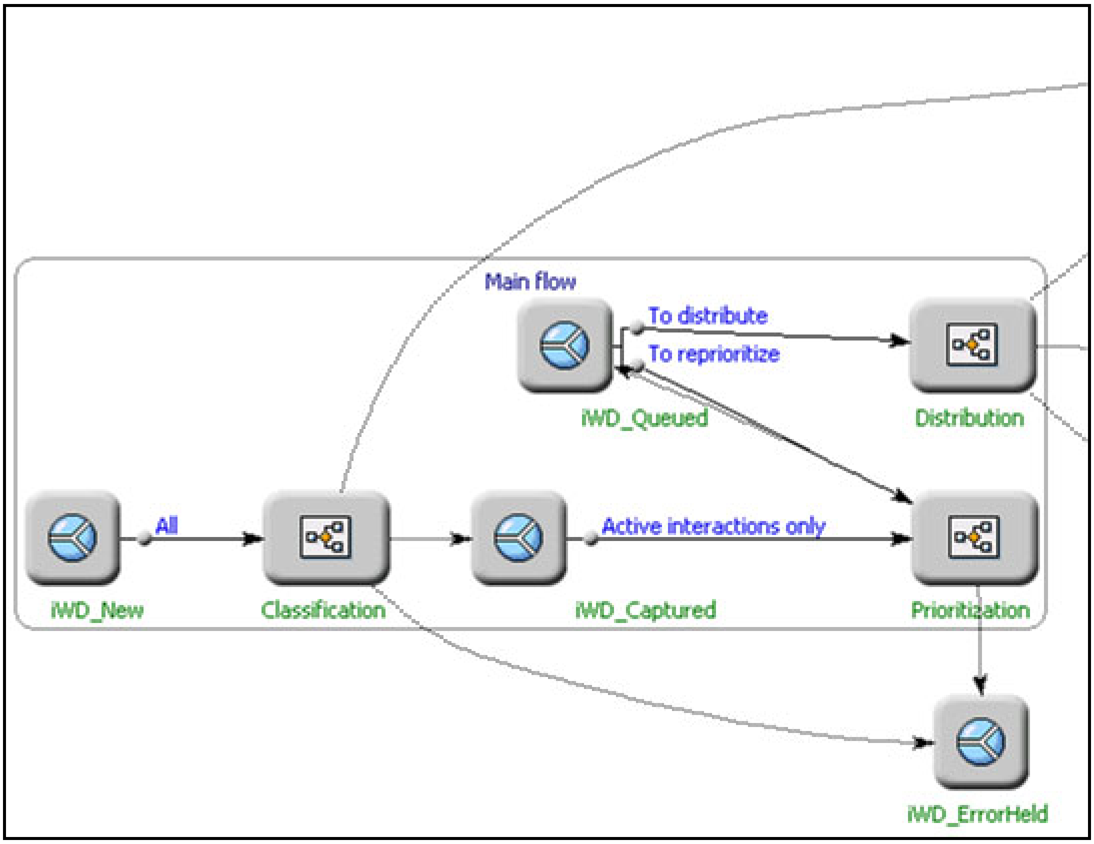


Figure 82: iWD Business Process—Main Flow

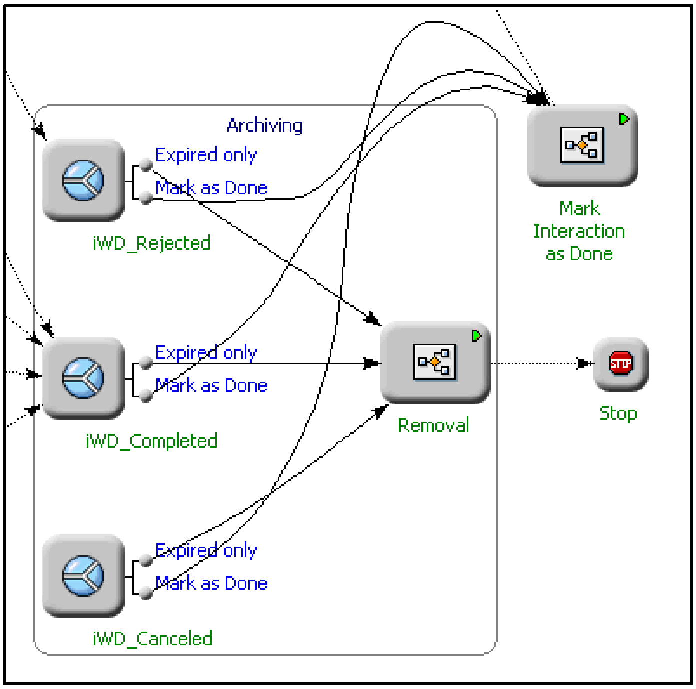


Figure 83: iWD Business Process—Archiving

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 84):

  - There are no conditions here.
  - Interactions are taken in order they were submitted.

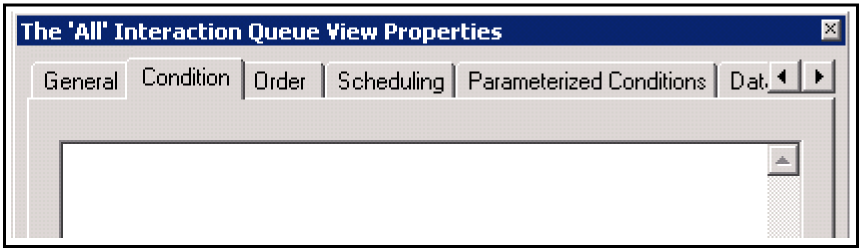


Figure 84: The ‘All’ Interaction Queue View Properties

Note: ESP stands for External Service Protocol. In this document it is the Business Context Management Service.

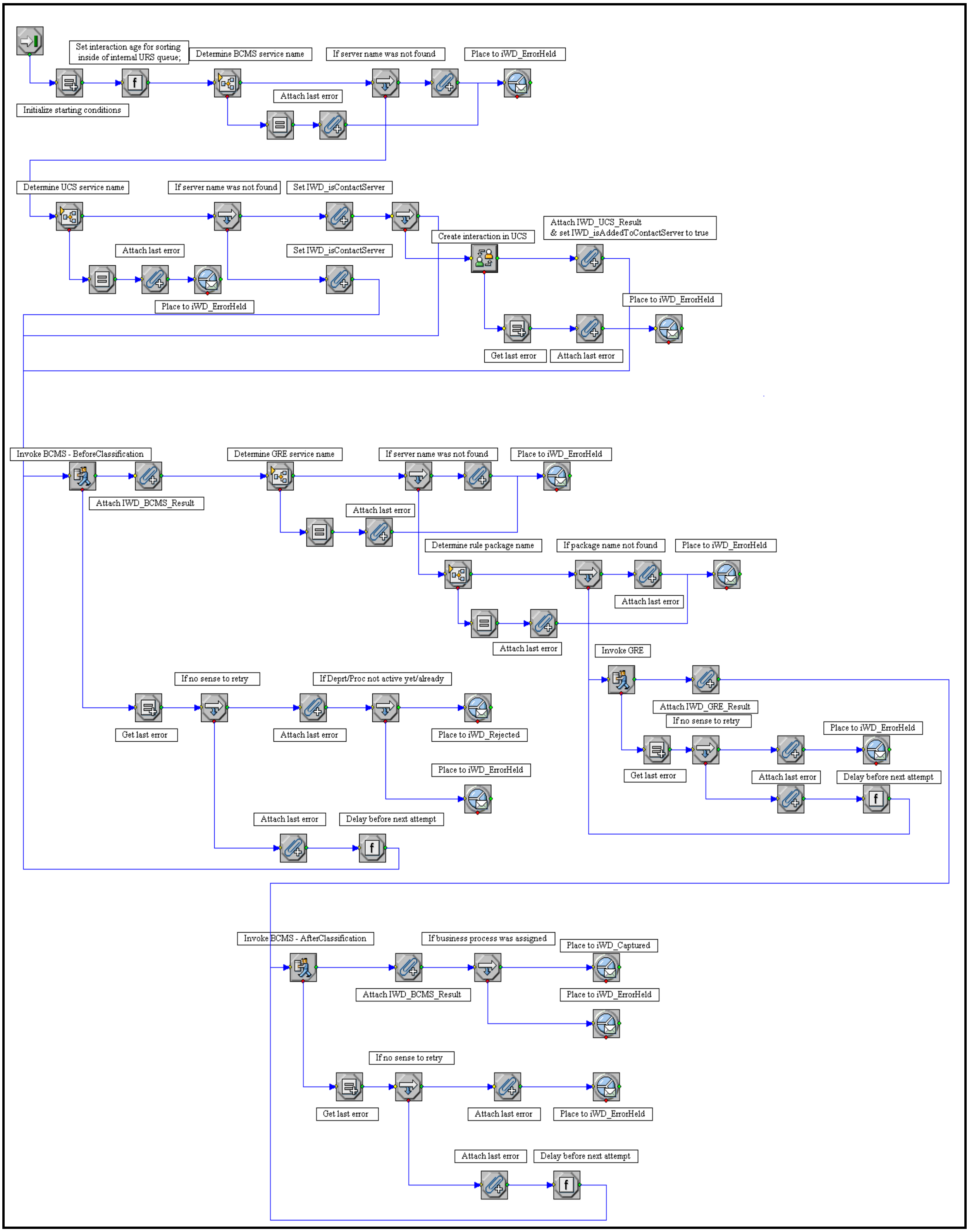


Figure 85: Classification Strategy (Entire Flow)

Summary of Flow

Steps 1 to 21 in this summary of flow are shown in Figure 86.

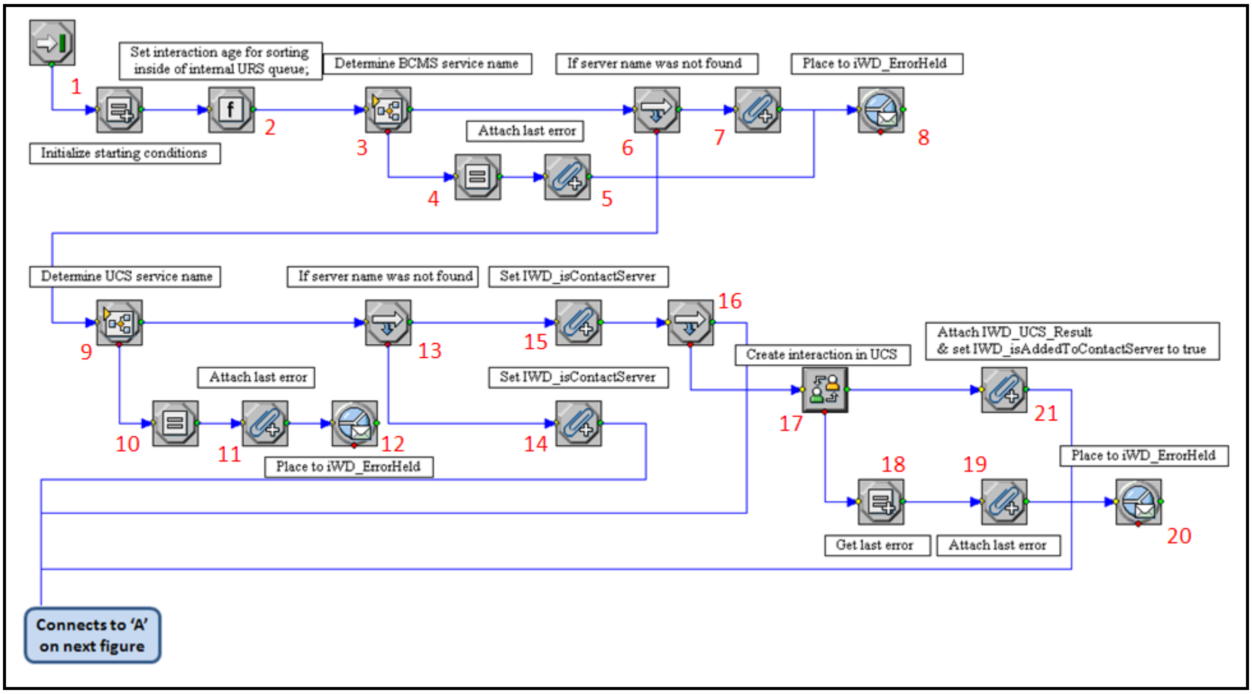
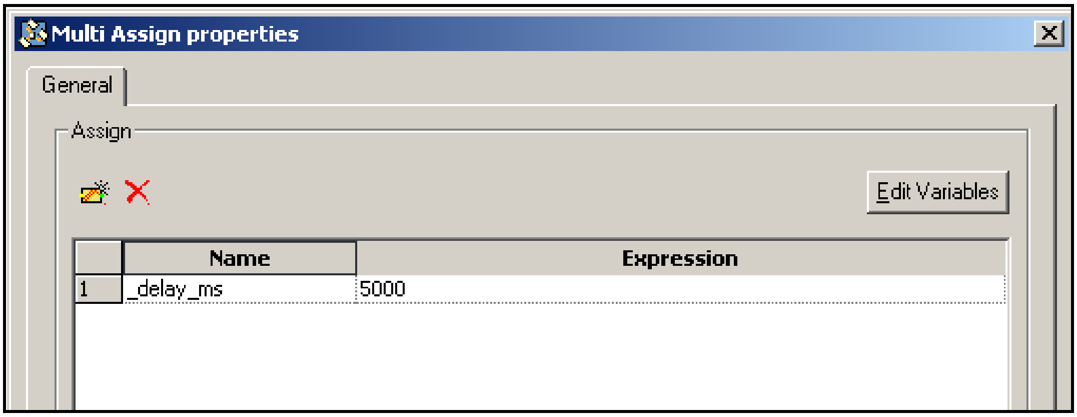

Set interaction age for sorting inside of internal URS queue;
Determine BCMS service name
If server name was not found
Place to iWD_ErrorHeld
1
2
3
6
7
8
Initialize starting conditions
Attach last error
4
5
Determine UCS service name
If server name was not found
Set IWD_isContactServer
16
9
13
15
Attach IWD_UCS_Result & set IWD_isAddedToContactServer to true
Create interaction in UCS
Attach last error
Set IWD_isContactServer
17
21
10
11
12
14
Place to iWD_ErrorHeld
18
19
Place to iWD_ErrorHeld
20
Get last error
Attach last error
Connects to 'A' on next figure

Figure 86: Classification Strategy—Section 1

1. A variable is initialized:
   - `_delay_ms` specifies the delay (in milliseconds) between attempts to invoke rules.



	Name	Expression
1	_delay_ms	5000

Figure 87: Initialize Variable—Classification Strategy

2. A command is sent to URS to use interaction age while sorting interactions in internal queues.

3. The DetermineESPServerName subroutine is invoked to determine the correct ESP server name to use. The subroutine uses the List Object list BCMSServerList. This subroutine also sets up cases when there is reason to retry to invoke the ESP server.
4. The subroutine fails an error is extracted.
5. This error is attached to user data as a key-value pair with the key IWD_BCMS_Determination_Error.
6. If the subroutine was successful, a check is done to ensure the existence of the ESP server name that was returned by the subroutine. If the ESP server name was found, the flow goes to 22.
7. If the ESP server name was not found, this error is attached to user data as a key-value pair with the key IWD_BCMS_Determination_Error.
8. The interaction is placed in the iWD_ErrorHeld queue.
9. The DetermineESPServerName subroutine is invoked to determine the correct ESP server name to use. The subroutine uses the List Object list called ContactServerList. This subroutine also sets up cases when there is reason to retry to invoke the ESP server.
10. If the subroutine fails an error is extracted.
11. This error is attached to user data as a key-value pair with the key IWD_UCS_Determination_Error.
12. The interaction is placed in the iWD_ErrorHeld queue.
13. If the subroutine was successful, a check is done to ensure the existence of the ESP server name that was returned by the subroutine. If the ESP server name was not found, the flow goes to 15.
14. The value of the user data key IWD_isContactServer is set to 0 (zero). The flow continues to 22.
15. The value of the user data key IWD_isContactServer is set to 1.
16. URS checks to see if the value of the user data key 'IWD_isAddedToContactServer is equal to 1, indicating that the task is already written into the interaction history in the UCS database. If the check evaluates as true, the flow continues to 22.
17. A new interaction is created in the UCS database, for this iWD task. If that function is successful, flow goes to 21.
18. If the creation of the interaction in UCS was unsuccessful, an error is extracted from user data.
19. This error is attached to user data as a key-value pair with the key IWD_UCS_Error.
20. The interaction is placed in the iWD_ErrorHeld queue.

21. The user data key `IWD_isAddedToContactServer` is updated to `1` to indicate that the task was successfully added to the interaction history in UCS. The result returned from the ESP call to UCS (from 17) is written to the variable `IWD_UCS_Result`.

Steps 22 to 48 in this summary of flow are shown in Figure 88 on page 324.

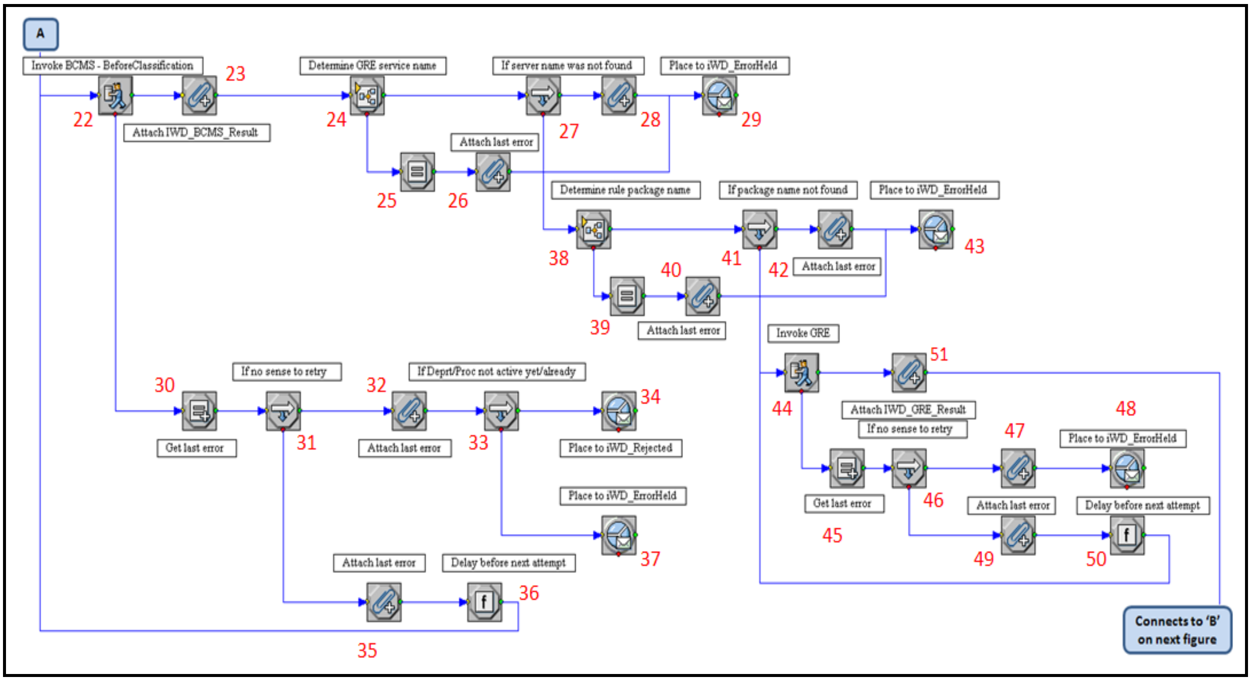


Figure 88: Classification Strategy—Section 2

22. A request is made to the ESP server, to prepare the interaction data before the Genesys Rules Engine can be called to invoke the classification rules.

23. If the communication with the ESP server was successful, the ESP result is attached to user data as a key-value pair with the key `IWD_BCMS_Result`. If not, the flow goes to 30.

24. The `DetermineESPServerName` subroutine is invoked to determine the name of the Genesys Rules Engine `Application`. The subroutine uses the `List Object` list `GREServerList`.

25. If the subroutine fails an error is extracted.

26. This error is attached to user data as a key-value pair with the key `IWD_GRE_Determination_Error`.

27. If the subroutine was successful, a check is done to ensure the existence of the ESP server name that was returned by the subroutine. If the ESP server name was found, the flow goes to 38.

28. If the ESP server name was not found, this error is attached to user data as a key-value pair with the key `IWD_GRE_Determination_Error`.

29. The interaction is placed in the `iWD_ErrorHeld` queue.
30. The last Interaction Server-related error is extracted from a variable.
31. A check is done to see if the error code is related to the ESP server communication.
32. The last error is attached to user data as a key-value pair with the key `IWD_BCMS_Error.`
33. A check is done to see if the error code is related to the iWD Department or Process not being available (for example, if the current date is outside of the Start and End Dates of the Department or Process).
34. If the Department or Process is not active yet, the interaction is placed in the `iWD_Rejected` queue.
35. The last error is attached to user data as a key-value pair with the key `IWD_BCMS_Error`.If not, the value of the `_counter` variable is incremented by 1.
36. A delay is introduced, based on the value of the variable `_delay_ms.` The flow goes back to 22 to retry the connection to the ESP server.
37. The interaction is placed in the `iWD_ErrorHeld` queue.
38. The `DetermineRulePackageName` subroutine is invoked to determine the name of the rule package that the Genesys Rules Engine will be invoking to evaluate the classification rules.
39. If the subroutine fails an error is extracted.
40. This error is attached to user data as a key-value pair with the key `IWD_Rule_Package_Determination_Error.`
41. If the subroutine was successful, a check is done to ensure the existence of the rule package name that was returned by the subroutine. If the rule package name was found, the flow goes to 44.
42. If the rule package name was not found, this error is attached to user data as a key-value pair with the key `IWD_Rule_Package_Determination_Error.`
43. The interaction is placed in the `iWD_ErrorHeld` queue.
44. An ESP request is sent to the Genesys Rules Engine to evaluate the classification rules.
45. The last Interaction Server-related error is extracted from a variable.
46. A check is done to see if the error code is related to the ESP server communication.
47. The last error is attached to user data as a key-value pair with the key `IWD_GRE_Error.`
48. The interaction is placed in the `iWD_ErrorHeld` queue.
49. The last error is attached to user data as a key-value pair with the key `IWD_GRE_Error`. If not, the value of the `_counter` variable is incremented by `1.`

50. A delay is introduced, based on the value of the _delay_ms variable. The flow goes back to 44 to retry the connection to the ESP server.

51. The result from the ESP call to the Genesys Rules Engine is attached to the interaction as user data, with the key IWD_GRE_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>, ...
```

    The following example shows what the result might look like:

```
AttributeUserData [list, size (unpacked)=168] =
'ESP_Result' [str] =
"return:ok|NumberOfRulesApplied:12|RulesApplied:McrSlt1GlbClsf1
McrSlt1GlbClassification1, McrSlt1GlbClsf2
McrSlt1GlbClassification2"
```

    The flow continues with step 52.

Steps 52 to 62 in this summary of flow are shown in Figure 89 on page 326.

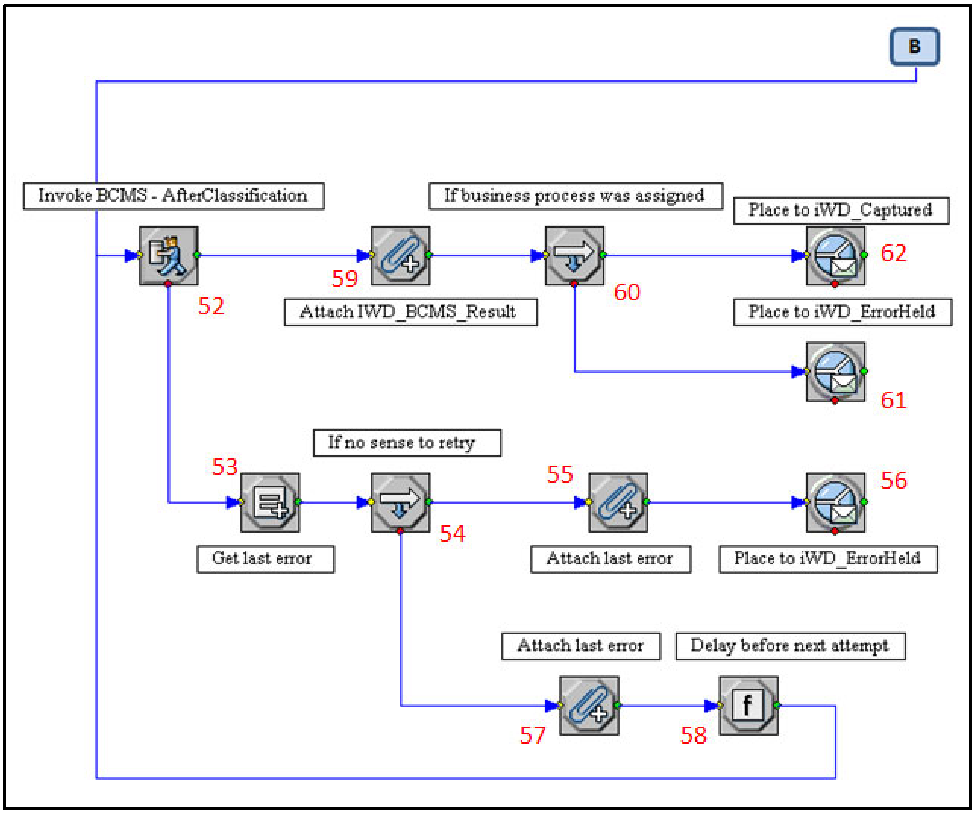


Figure 89: Classification Strategy—Section 3

52. A request is made to the ESP server, to ensure the integrity of the interaction data that was returned after the rules were invoked by the Genesys Rules Engine.
53. The last Interaction Server-related error is extracted from a variable.
54. A check is done to see if the error code is related to the ESP server communication.
55. The last error is attached to user data as a key-value pair with the key `IWD_BCMS_Error`.
56. The interaction is placed in the `iWD_ErrorHeld` queue.
57. If the error check in 54 determined that the last error was potentially communication-related, the last error is attached to user data as a key-value pair with the key `IWD_BCMS_Error`.
58. A delay is introduced, based on the value of the `_delay_ms` variable. The flow goes back to 52 to retry the connection to the ESP server.
59. The ESP result is attached to user data as a key-value pair with the key `IWD_BCMS_Result`.
60. Verification is done to check if a business process was assigned by a classification rule.
61. If no business process was assigned, the interaction is placed into the `iWD_ErrorHeld` queue.
62. If a business process was assigned, then the interaction is placed in the `iWD_Captured` queue.

Prioritization Strategy

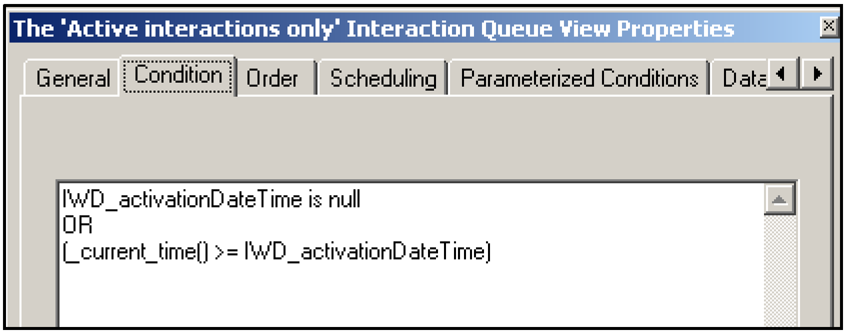
The purpose of this strategy is to invoke the corresponding prioritization rules, analyze the result of the rules application and place the interaction into the appropriate queue, depending on the result.

This strategy processes interactions from the following queues:

- `iWD_Captured`

  Interactions have to satisfy following conditions (see Figure 90):

  - 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.


The 'Active interactions only' Interaction Queue View Properties
General
Condition
Order
Scheduling
Parameterized Conditions
Data

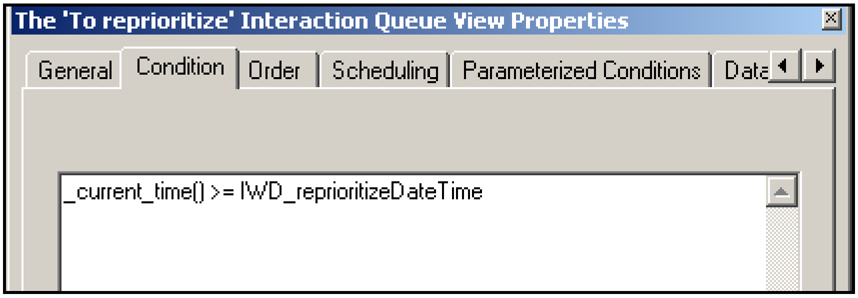
```
IWD_activationDateTime is null
OR
(_current_time() >= IWD_activationDateTime)
```

Figure 90: The 'Active interactions only' Interaction Queue View Properties

- `iWD_Queued`

  Interactions have to satisfy the following conditions (see Figure 91):

  - Interactions that are subject for immediate reprioritization (interactions that have the property `IWD_reprioritizeDateTime` set to a time stamp which is in the past)
  - Interactions are taken in order of `IWD_reprioritizationDateTime` (oldest first).



```
_current_time() >= IWD_reprioritizeDateTime
```

Figure 91: The 'To reprioritize' Interaction Queue View Properties

Figure 92 shows the `Prioritization` strategy.

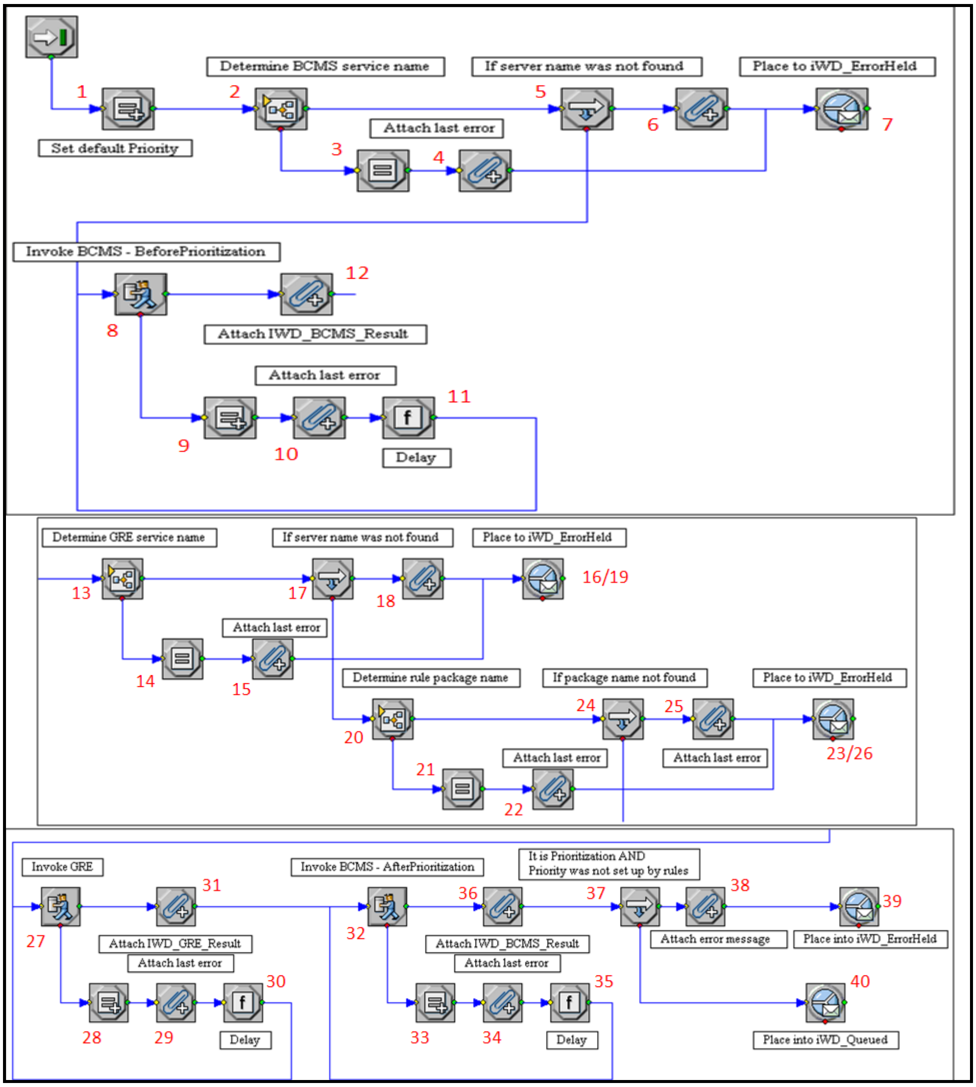


Figure 92: Prioritization Strategy (Entire Flow)

Summary of Flow

Steps 1 to 12 in this summary of flow are shown in Figure 93.

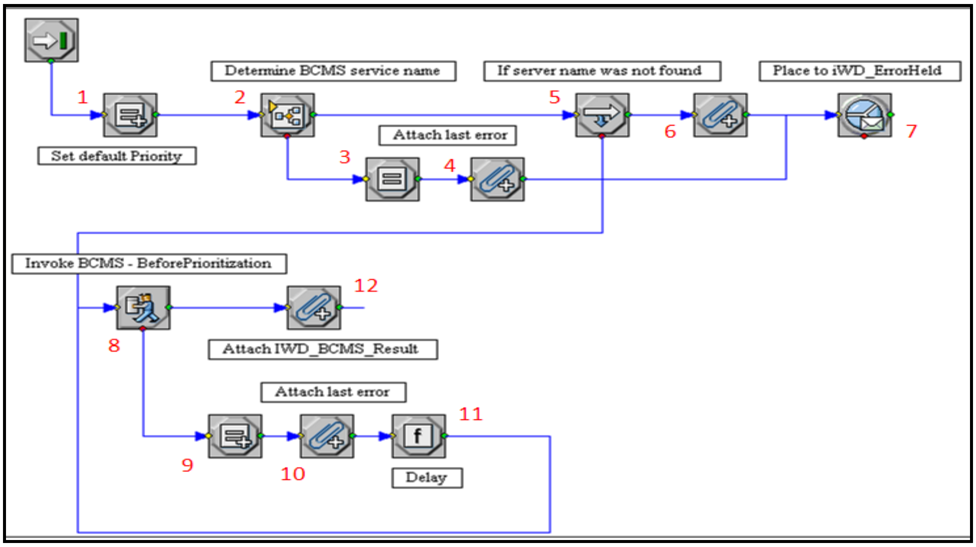


Figure 93: Prioritization Strategy—Section 1

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.
   - `_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 `DetermineESPServerName` subroutine is invoked to determine the correct ESP server name to use. The subroutine uses the `List Object` list `BCMSServerList`. 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_BCMS_Determination_Error`.
5. If the subroutine was successful, a check is done to ensure the existence of the ESP server name that was returned by the subroutine.
6. If the ESP server name was not found, this error is attached to user data as a key-value pair with the key `IWD_BCMS_Determination_Error`.
7. The interaction is placed in the `iWD_ErrorHeld` queue.
8. A request is made to the ESP server, to prepare the interaction data before the Genesys Rules Engine can be called to invoke the prioritization rules.
9. The last Interaction Server-related error is extracted from a variable.
10. The last error is attached to user data as a key-value pair with the key `IWD_BCMS_Error`.

11. A delay is introduced, based on the value of the `_error_timeout_ms` variable. The flow goes back to 8 to retry the connection to the ESP server.
12. The ESP result is attached to user data as a key-value pair with the key `IWD_BCMS_Result`.

Steps 13 to 26 in this summary of flow are shown in Figure 94.

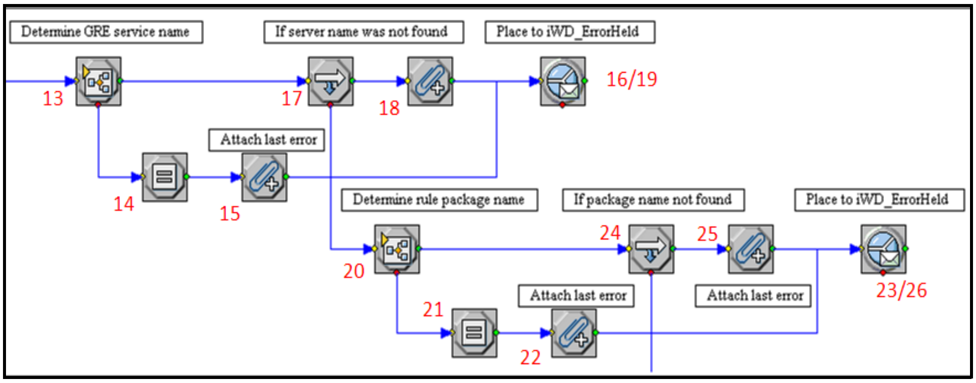


Figure 94: Prioritization Strategy—Section 2

13. The `DetermineESPServerName` subroutine is invoked to determine the name of the Genesys Rules Engine application. The subroutine uses the `List Object` list `GREServerList`.
14. If the subroutine fails an error is extracted.
15. This error is attached to user data as a key-value pair with the key `IWD_GRE_Determination_Error`.
16. The interaction is placed in the `iWD_ErrorHeld` queue.
17. If the subroutine was successful, a check is done to ensure the existence of the ESP server name that was returned by the subroutine. If the ESP server name was found, flow goes to 20.
18. If the ESP server name was not found, this error is attached to user data as a key-value pair with the key `IWD_GRE_Determination_Error`.
19. The interaction is placed in the `iWD_ErrorHeld` queue.
20. The `DetermineRulePackageName` subroutine is invoked to determine the name of the rule package that the Genesys Rules Engine will be invoking to evaluate the prioritization rules.
21. If the subroutine fails an error is extracted.
22. This error is attached to user data as a key-value pair with the key `IWD_Rule_Package_Determination_Error`.
23. The interaction is placed in the `iWD_ErrorHeld` queue.
24. If the subroutine was successful, a check is done to ensure the existence of the rule package name that was returned by the subroutine. If the rule package name was found, flow goes to 27.

25. If the rule package name was not found, this error is attached to user data as a key-value pair with the key `IWD_Rule_Package_Determination_Error`.
26. The interaction is placed in the `iWD_ErrorHeld` queue.

Steps 27 to 40 in this summary of flow are shown in Figure 95.

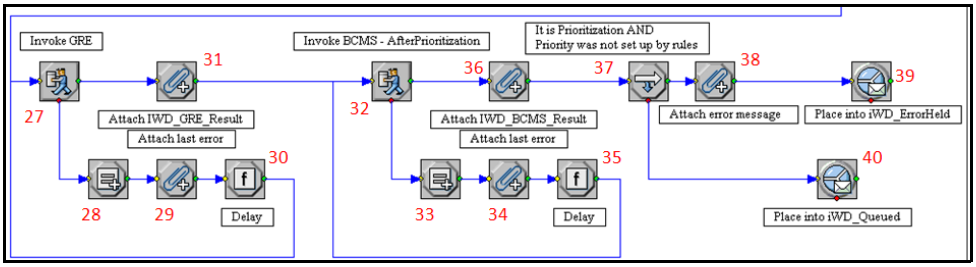


Figure 95: Prioritization Strategy—Section 3

27. An ESP request is sent to the Genesys Rules Engine to evaluate the prioritization rules. If the request to the ESP server was successful, flow goes to 31.
28. The last Interaction Server-related error is extracted from a variable.
29. The last error is attached to user data as a key-value pair with the key `IWD_GRE_Error`.
30. A delay is introduced, based on the value of the `_error_timeout_ms` variable. The flow goes back to 27 to retry the connection to the ESP server.
31. If the ESP server reports that the operation was completed successfully, the results are attached to user data as a key-value pair with the key `IWD_GRE_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>, "
    ```

    The following is an example of what the result might look like:

    ```
    AttributeUserData [list, size (unpacked)=168] =
    'ESP_Result' [str] =
    "return:ok|NumberOfRulesApplied:2|RulesApplied:McrSlt1GlbPrior1
    McrSlt1GlbPrioritization1, McrSlt1GlbClsf2
    McrSlt1GlbPrioritization2"
    ```

32. A request is made to the ESP server, to ensure the integrity of the interaction data that was returned after the rules were invoked by the Genesys Rules Engine. If the request was successful, flow goes to 36.
33. If the request to the ESP server was not successful, the last Interaction Server-related error is extracted from a variable.
34. The last error is attached to user data as a key-value pair with the key `IWD_BCMS_Error`.

35. A delay is introduced, based on the value of the `_error_timeout_ms` variable. The flow goes back to 32 to retry the connection to the ESP server.
36. The ESP result is attached to user data as a key-value pair with the key `IWD_BCMS_Result`.
37. A check is made to see if this is the first time that prioritization rules are being evaluated for the interaction, and the priority was not set up by any rules. If this check is false, flow goes to 40.
38. The error message Priority is not set up by rules is attached to interaction server data as a key-value pair with the key `IWD_Prioritization_Error`.
39. The interaction is placed in the `iWD_ErrorHeld` queue.
40. The interaction is placed in the `iWD_Queued` queue.

Distribution Strategy

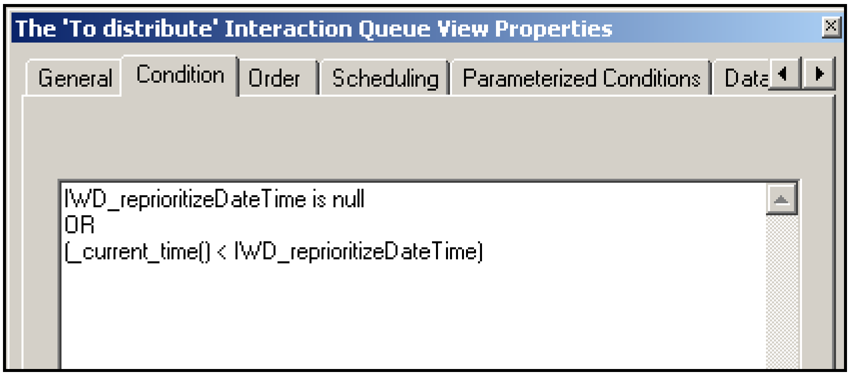
This simple strategy routes interactions to a requested Agent, requested Agent Group, requested Skill, or to the default iWD Agent Group.

This strategy processes interactions from the following queues:

- `iWD_Queued`

  Interactions have to satisfy the following conditions (see Figure 96):

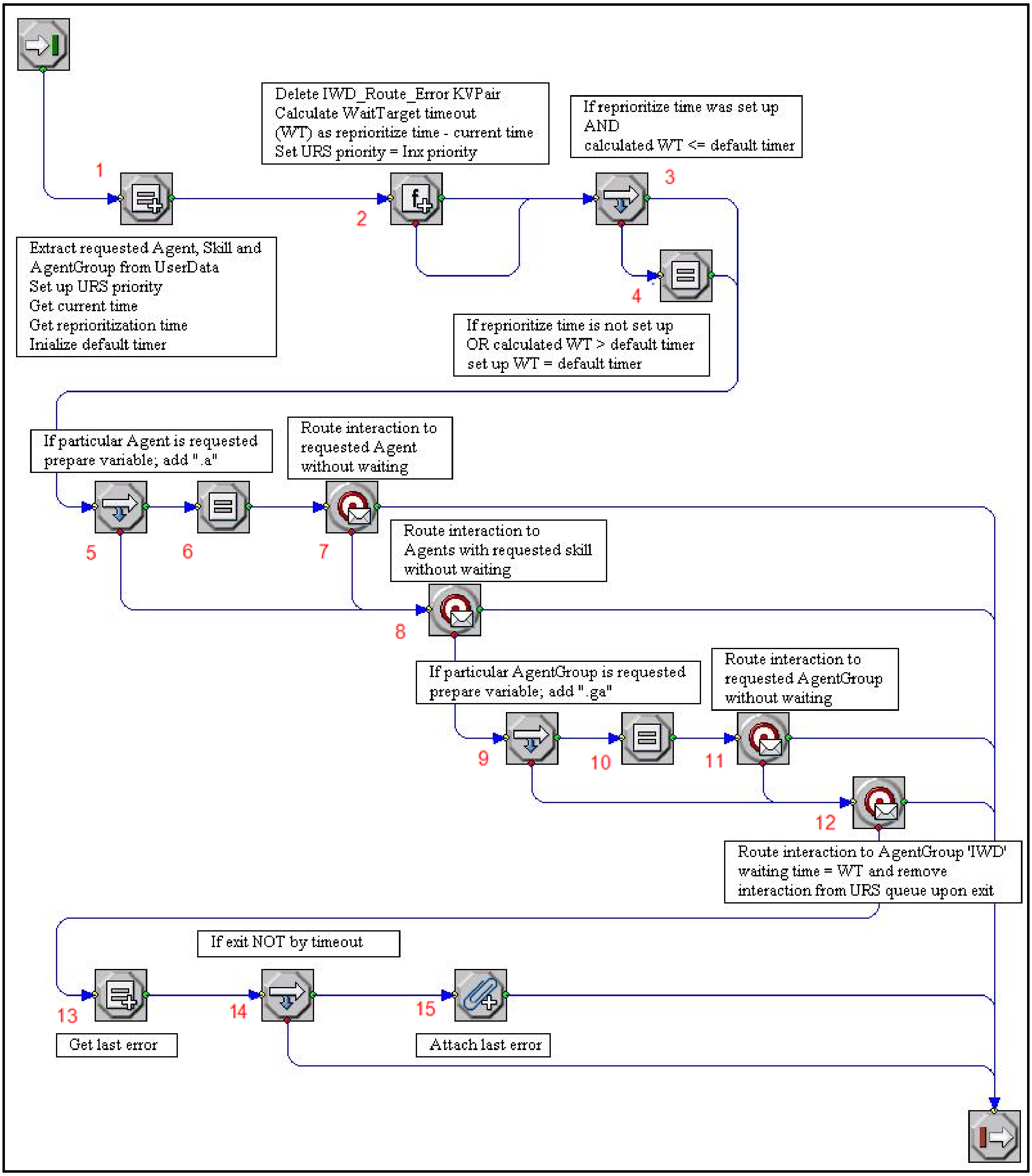
  - Interactions that are not subject for immediate reprioritization (interactions that do not have the property `IWD_reprioritizeDateTime` set, or that have this property set to a time stamp that is in the future).
  - Interactions are taken in order of priority (highest priority first)



```
IWD_reprioritizeDateTime is null
OR
(_current_time() < IWD_reprioritizeDateTime)
```

Figure 96: The 'To distribute' Interaction Queue View Properties

Figure 97 shows the `Distribution` strategy.



Delete IWD_Route_Error KVPair
Calculate WaitTarget timeout
(WT) as reprioritize time - current time
Set URS priority = Inx priority

If reprioritize time was set up
AND
calculated WT <= default timer

Extract requested Agent, Skill and
AgentGroup from UserData
Set up URS priority
Get current time
Get reprioritization time
Inialize default timer

If reprioritize time is not set up
OR calculated WT > default timer
set up WT = default timer

If particular Agent is requested
prepare variable; add ".a"

Route interaction to
requested Agent
without waiting

Route interaction to
Agents with requested skill
without waiting

If particular AgentGroup is requested
prepare variable; add ".ga"

Route interaction to
requested AgentGroup
without waiting

Route interaction to AgentGroup 'IWD'
waiting time = WT and remove
interaction from URS queue upon exit

If exit NOT by timeout

Get last error

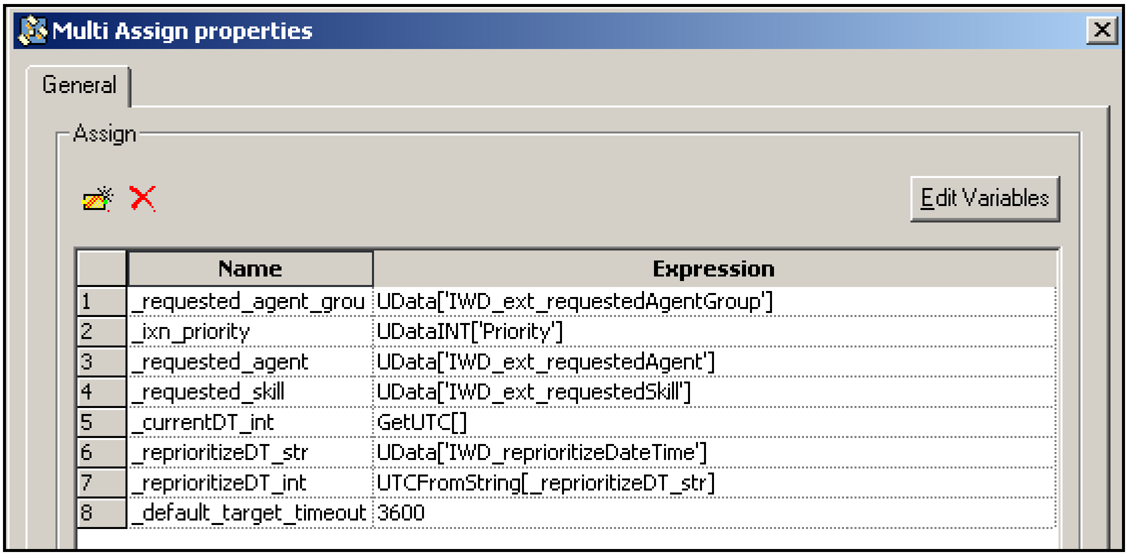
Attach last error

Figure 97: Distribution Strategy

Summary of Flow

Note: The IRD objects described in this section are keyed to the numbers in Figure 97.

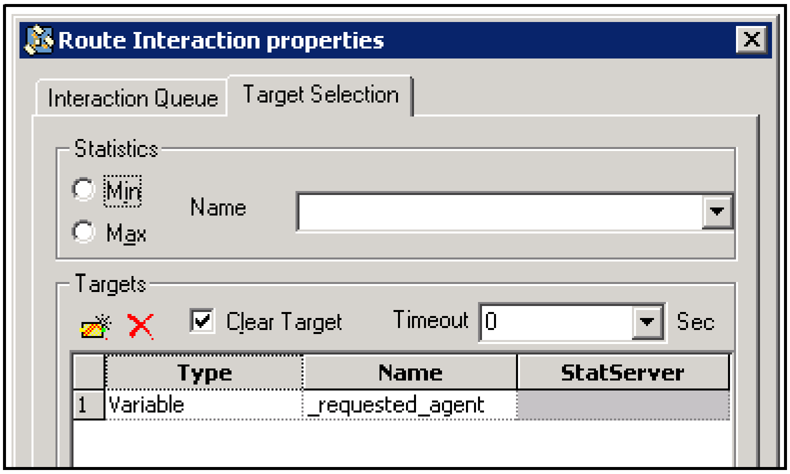
1. Extract information about requested agent, agent group, or skill and initialize internal variables. See Figure 98.



	Name	Expression
1	_requested_agent_grou	UData['IWD_ext_requestedAgentGroup']
2	_ixn_priority	UDataINT['Priority']
3	_requested_agent	UData['IWD_ext_requestedAgent']
4	_requested_skill	UData['IWD_ext_requestedSkill']
5	_currentDT_int	GetUTC[]
6	_reprioritizeDT_str	UData['IWD_reprioritizeDateTime']
7	_reprioritizeDT_int	UTCFromString[_reprioritizeDT_str]
8	_default_target_timeout	3600

Figure 98: 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 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 99).



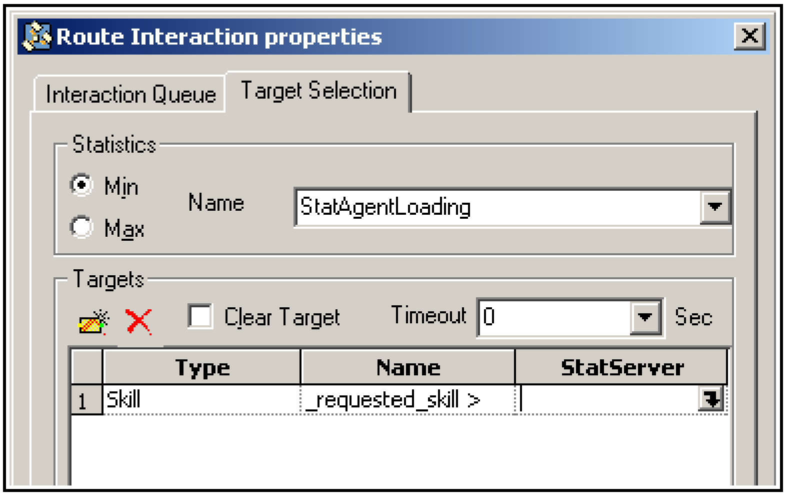
○ Min
○ Max
Name

☑ Clear Target Timeout 0 Sec

	Type	Name	StatServer
1	Variable	_requested_agent	

Figure 99: Route to Agent

8. Try to route the interaction to an agent with the requested skill without waiting (see Figure 100).



● Min
○ Max
Name StatAgentLoading

□ Clear Target Timeout 0 Sec

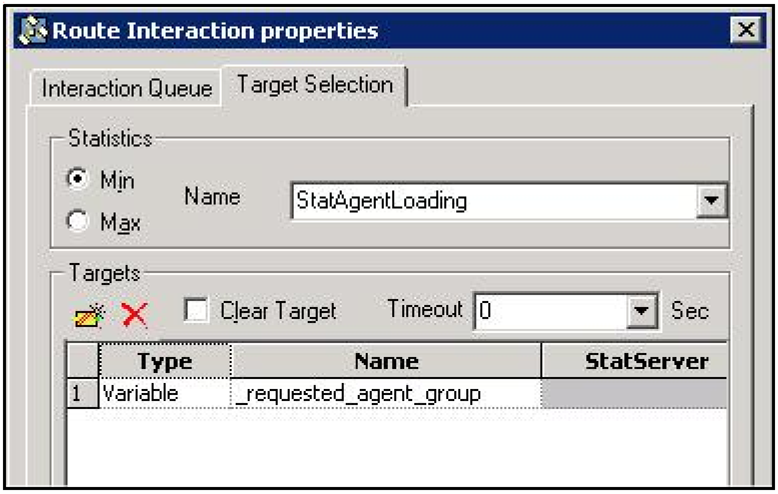
	Type	Name	StatServer
1	Skill	_requested_skill >	

Figure 100: 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 101).



Statistics

● Min ○ Max Name StatAgentLoading

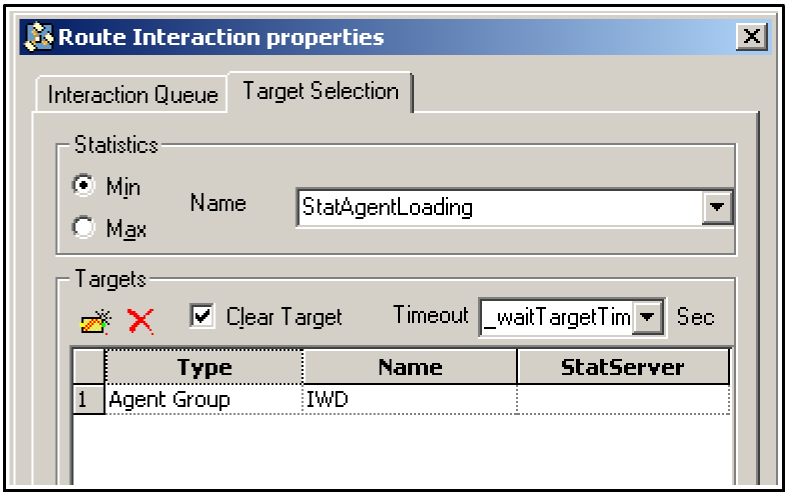
Targets

□ Clear Target Timeout 0 Sec

	Type	Name	StatServer
1	Variable	_requested_agent_group	

Figure 101: 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 102).



Statistics

● Min ○ Max Name StatAgentLoading

Targets

☑ Clear Target Timeout _waitTargetTim Sec

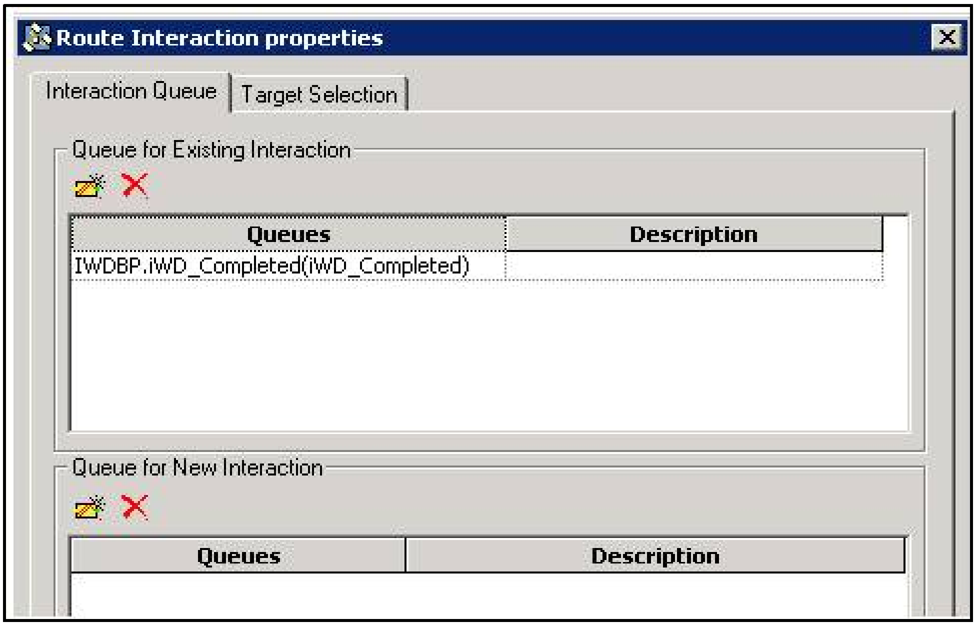
	Type	Name	StatServer
1	Agent Group	IWD	

Figure 102: 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 8.1 Reference Manual*.

The Route Interaction object also has an `Interaction Queue` tab. See Figure 103. (This applies to all three Route Interaction objects in this strategy.)


Route Interaction properties
Interaction Queue
Target Selection
Queue for Existing Interaction
Queue for New Interaction

Queues	Description
IWDBP.iWD_Completed(iWD_Completed)	

Queues	Description

Figure 103: 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).

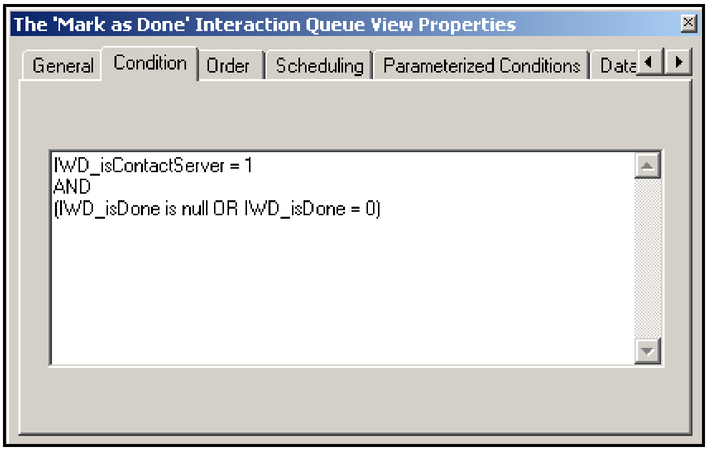
A Description (optional) appears as a hint on the agent desktop as to where to place the interaction.

Mark Interaction as Done Strategy

The purpose of this strategy is to update the Universal Contact Server (UCS) database to mark the interaction as *done*. This equates to setting the value in the `Status` column of the `Interactions` table to `3`. UCS clients, such as Interaction Workspace, will then display the status of this interaction as *done* when the user looks at interactions they have previously processed.

Interactions have to satisfy the following conditions (see Figure 104 on page 339):

- The value of the attached data key `IWD_isContactServer` is `1`
- The value of the attached data key `IWD_isDone` is either null or `0` (zero)



```
IWD_isContactServer = 1
AND
(IWD_isDone is null OR IWD_isDone = 0)
```

Figure 104:Mark Interaction as Done View

This strategy processes interactions from the following queues:

- `iWD_Completed`
- `iWD_Canceled`
- `iWD_Rejected`

Figure 105 shows the `Mark Interaction as Done` strategy.

Summary of Flow

Steps 1 to 16 in this summary of flow are shown in Figure 105 on page 339.

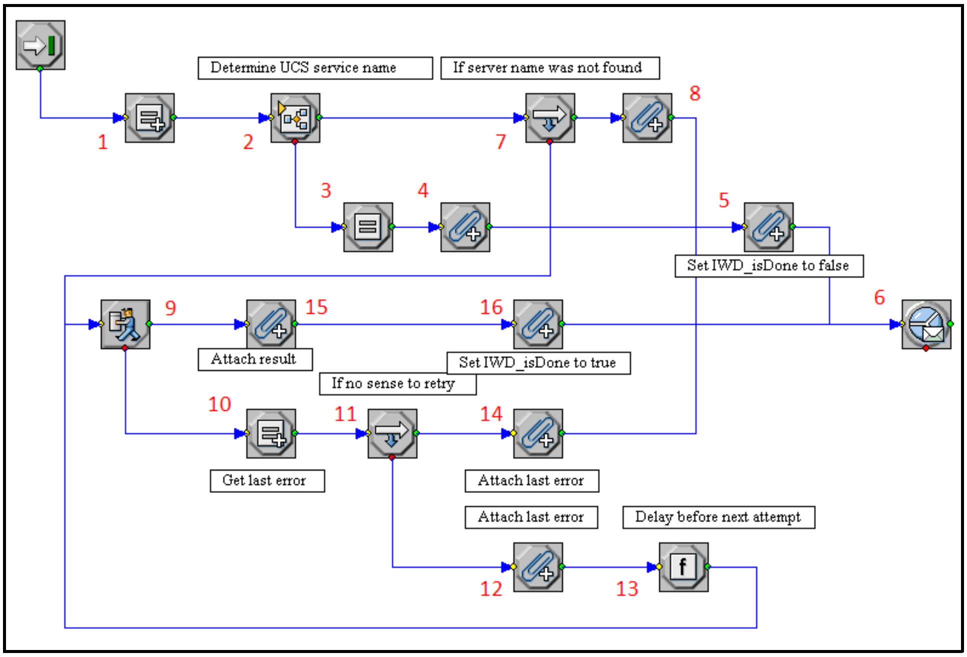


Figure 105:Mark Interaction as Done Strategy

1. Variables are initialized:
   - `_tenant_id` is the Configuration Server Tenant ID associated with the interaction being processed by this strategy.
   - `_interaction_id` is the ID of the interaction being processed by this strategy.
   - `_delay_ms` specifies the delay (in milliseconds) between attempts to communicate with the Universal Contact Server.
2. The `DetermineESPServerName` subroutine is invoked to determine the correct ESP server name to use. The subroutine uses the List Object list called `ContactServerList`. This subroutine also sets up cases when there is reason to retry to invoke the ESP server. If the subroutine is successful, the flow continues to 7.
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_UCS_Determination_Error`.
5. The value of the user data key `IWD_isDone` is set to `0` (zero).
6. The interaction is returned to its previous queue.
7. If the subroutine was successful, a check is done to ensure the existence of the ESP server name that was returned by the subroutine. If the ESP server name was found, the flow goes to 9.
8. If the ESP server name was not found, this error is attached to user data as a key-value pair with the key `IWD_UCS_Determination_Error`. Flow goes to step 5.
9. The strategy calls a method on the Universal Contact Server to set the status of the interaction to `3`, indicating that the interaction is *done*.
10. If the invocation of the method on the Universal Contact Server fails, an error is extracted.
11. A check is done to evaluate the error code extracted in step 10.
12. If it makes sense to retry updating the interaction record in UCS, the last error code is attached to the interaction with the user data key `IWD_UCS_Error`.
13. A delay is introduced into the processing. Flow returns to step 9.
14. If it does not make sense to retry updating the interaction record in UCS, the last error is attached to the interaction with the user data key `IWD_UCS_Error`. Flow goes to step 5.
15. If the UCS update in step 9 was successful, the result from UCS is attached to the interaction with user data key `IWD_UCS_Result`.
16. The value of the user data key `IWD_isDone` is set to `1`. Flow goes to step 6.

Removal Strategy

The purpose of this strategy is to delete expired interactions from the Interaction Server database.

Note: This routing strategy has changed significantly from iWD 8.0 and 8.1.0, where it was called the `Archive` strategy. Please see "Task Archiving" on page 21.

A key-value pair in user data with the key `IWD_expirationDateTime` contains information about when an interaction has to be `deleted`.

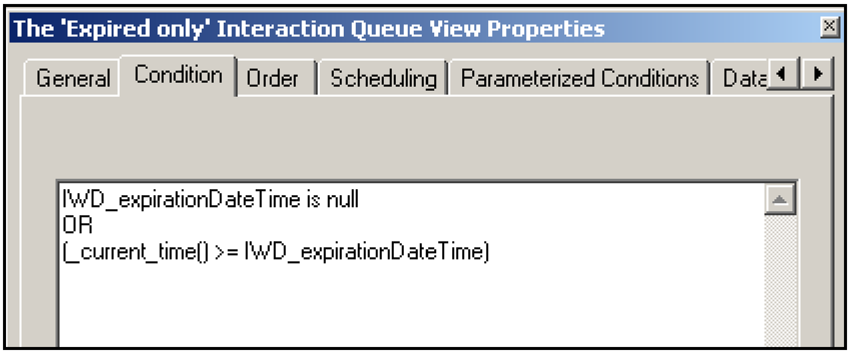
Note: In release 8.1.1, the meaning of parameter `IWD_expirationDateTime` has changed from previous releases. `IWD_expirationDateTime` in 8.1.1 defines the amount of time for which a task is going to be kept in the Interaction Server database.

This strategy processes interactions from the following queues:

- `iWD_Completed`
- `iWD_Canceled`
- `iWD_Rejected`

Interactions have to satisfy the following conditions (see Figure 106):

- Interactions must either have the property `IWD_expirationDateTime` set, or this property must have a time stamp which is in the past.
- Interactions are taken in the order they were submitted.



```
IWD_expirationDateTime is null
OR
(_current_time() >= IWD_expirationDateTime)
```

Figure 106: The 'Expired only' Interaction Queue View Properties

Figure 107 shows the `Removal` strategy.

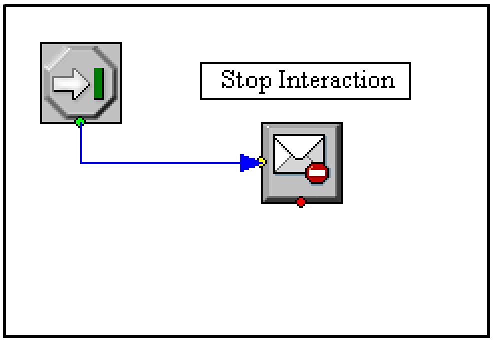


Figure 107:Removal Strategy (Entire Flow)

Summary of Flow

When interactions enter the Removal strategy, the processing of the interaction is stopped. This means that the interaction is deleted from the Interaction Server database.

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 333). 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).

You can also add additional interaction queues into the IWDBP business process, based on your business requirements. However, keep the following points in mind:

- The `iWD_Queued` queue must be present for Data Mart to properly count interactions/tasks. You can add other queues to the business process, but only after interactions have passed through the `iWD_Queued` queue.
- Data Mart can properly determine when to consider a task as complete, only if the final queue in the business process is one of the following: `iWD_Rejected`, `iWD_Canceled`, or `iWD_Completed`.

Appendix

D 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 social media, 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

This section explains what should be done to adapt the default 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 (`iWD_New`) only *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 108). 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`.

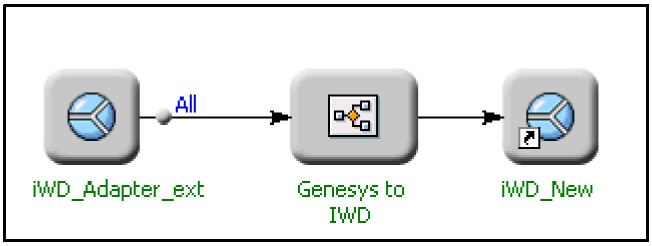


Figure 108: Standard Genesys to iWD Adapter Business Process

Figure 109 shows the `Genesys to IWD` strategy.

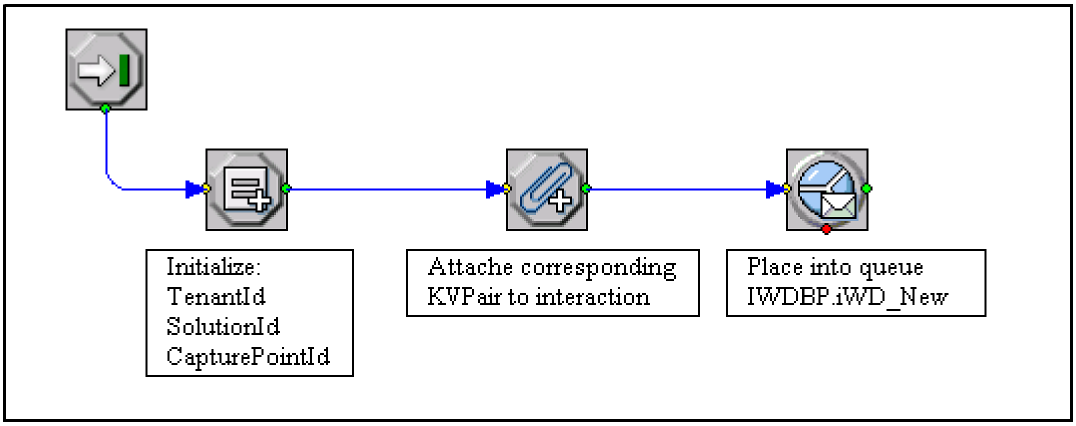
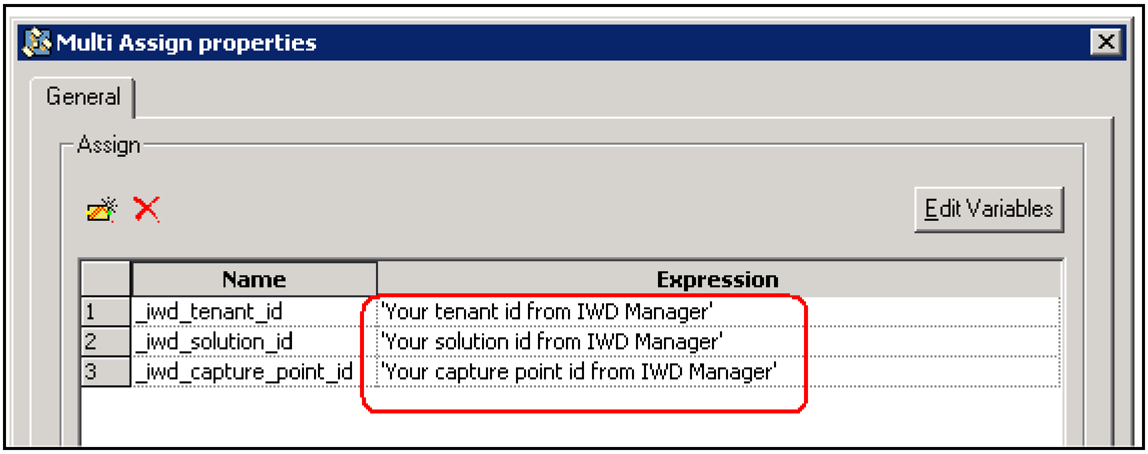

Initialize:
TenantId
SolutionId
CapturePointId
Attache corresponding KVPair to interaction
Place into queue IWDBP.iWD_New

Figure 109: Genesys to iWD Strategy

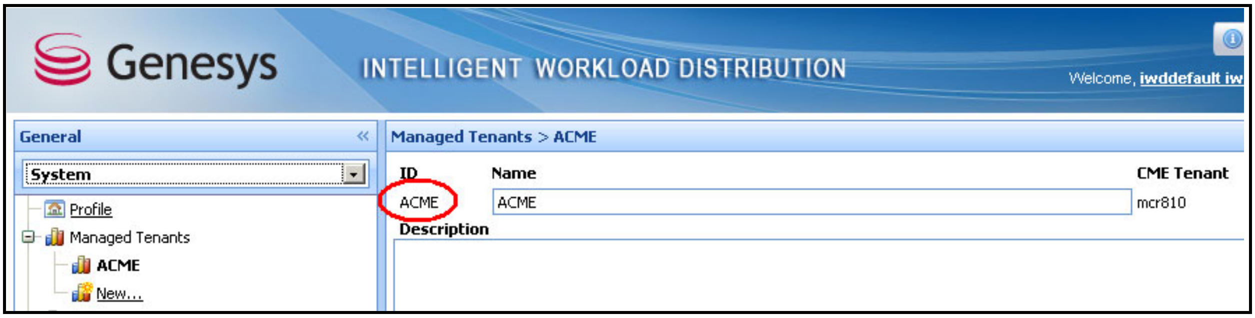
In the `Multi Assign` object, you have to initialize all variables, as shown in Figure 110 on page 345.



	Name	Expression
1	_iwd_tenant_id	'Your tenant id from IWD Manager'
2	_iwd_solution_id	'Your solution id from IWD Manager'
3	_iwd_capture_point_id	'Your capture point id from IWD Manager'

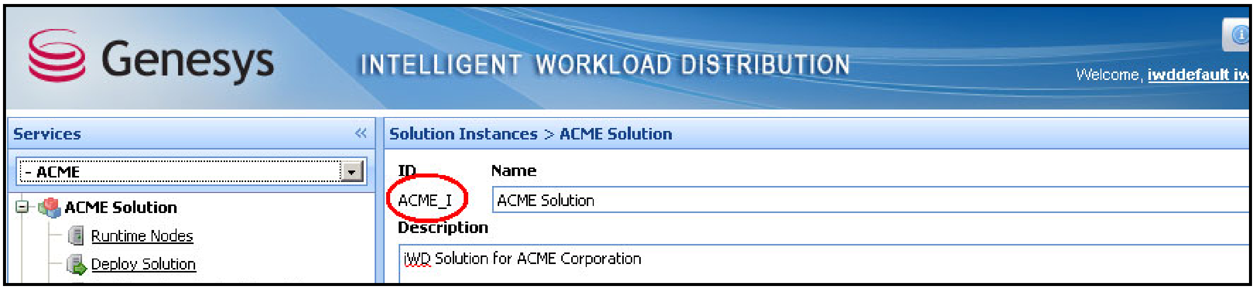
Figure 110: Assign Variables

The IDs are taken from iWD Manager. Figure 111 and Figure 112 show where to find your Tenant ID and Solution ID, respectively.



ID: ACME
Name: ACME
CME Tenant: mcr810
Description:

Figure 111: Tenant ID



ID: ACME_I
Name: ACME Solution
Description: iWD Solution for ACME Corporation

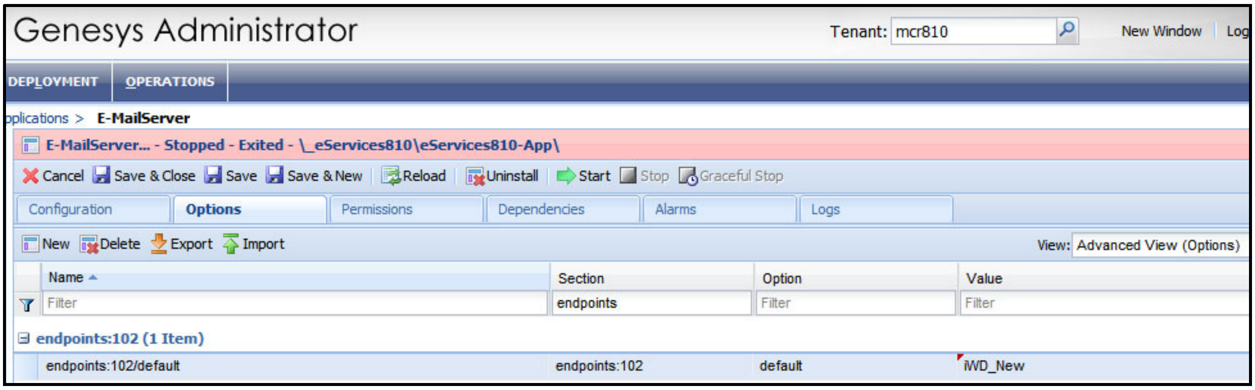
Figure 112: Solution ID

To get an ID for a capture point, you have to configure a Generic Capture Point service (see “Generic Capture Point Service” on page 188). The ID of the Generic Capture Point service must be populated in the iWD_CapturePointId user data key in the Genesys to iWD routing strategy that was described earlier. It will represent a Genesys standard server (in our example, the Genesys E-mail Server).

Adding E-mail Server

A Genesys E-mail Server needs to be added to the Standard Genesys to IWD Adapter business process. You can add the 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 113. Next, refresh in IRD and E-mail Server will be added to the business process with the iWD_Adapter_ext queue.



E-MailServer... - Stopped - Exited - _eServices810\eServices810-App\

Name	Section	Option	Value
Filter	endpoints	Filter	Filter
endpoints:102 (1 Item)			
endpoints:102/default	endpoints:102	default	iWD_New

Figure 113: 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 114 shows how the Standard Genesys to IWD Adapter business process will look after these modifications.

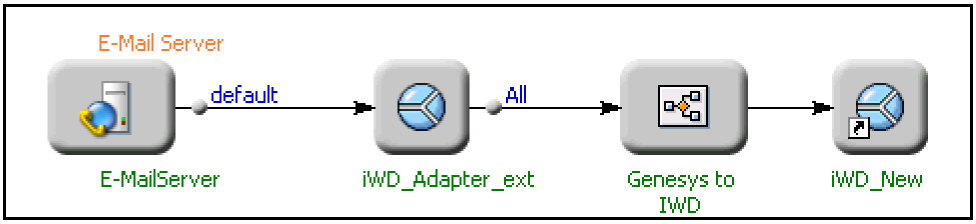
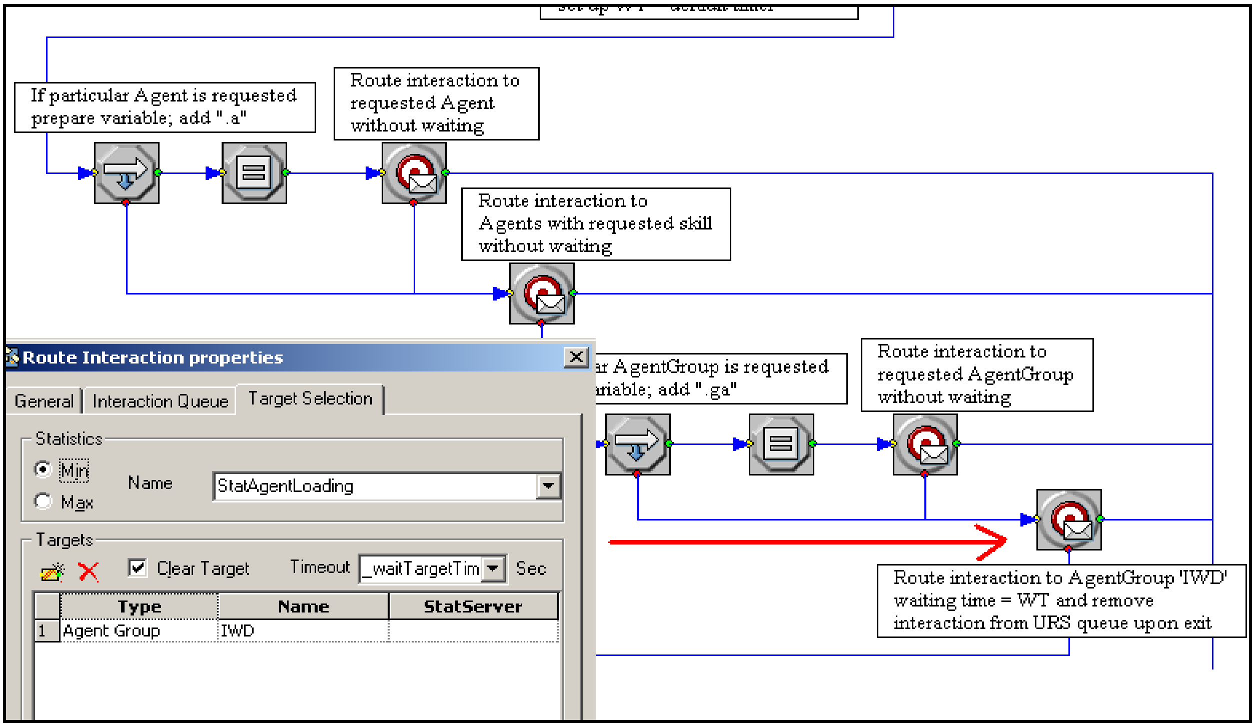


Figure 114: Standard Genesys to iWD Adapter Business Process After Modifications

Modify the Distribution Strategy

In IRD, open the IWDBP business process. Open the Distribution strategy. Since there are business actions Request Agent, Request Agent Group, and Request Skill, the default business process has to take three into consideration. That is why there are four 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. If you want to change the length of time URS has to wait for the next available agent (by default it is set to 60 seconds), you can do so by changing the initial value of the `variable _default_target_timeout` in the first `MultiSelect` block in the routing strategy. See Figure 115.


If particular Agent is requested prepare variable; add ".a"
Route interaction to requested Agent without waiting
Route interaction to Agents with requested skill without waiting
...ar AgentGroup is requested ...riable; add ".ga"
Route interaction to requested AgentGroup without waiting
Route interaction to AgentGroup 'IWD' waiting time = WT and remove interaction from URS queue upon exit

Route Interaction properties

General | Interaction Queue | Target Selection

Statistics

● Min ○ Max Name: StatAgentLoading

Targets

☑ Clear Target Timeout: _waitTargetTim Sec

	Type	Name	StatServer
1	Agent Group	IWD	

Figure 115: Update Route Interaction to Agent Group IWD

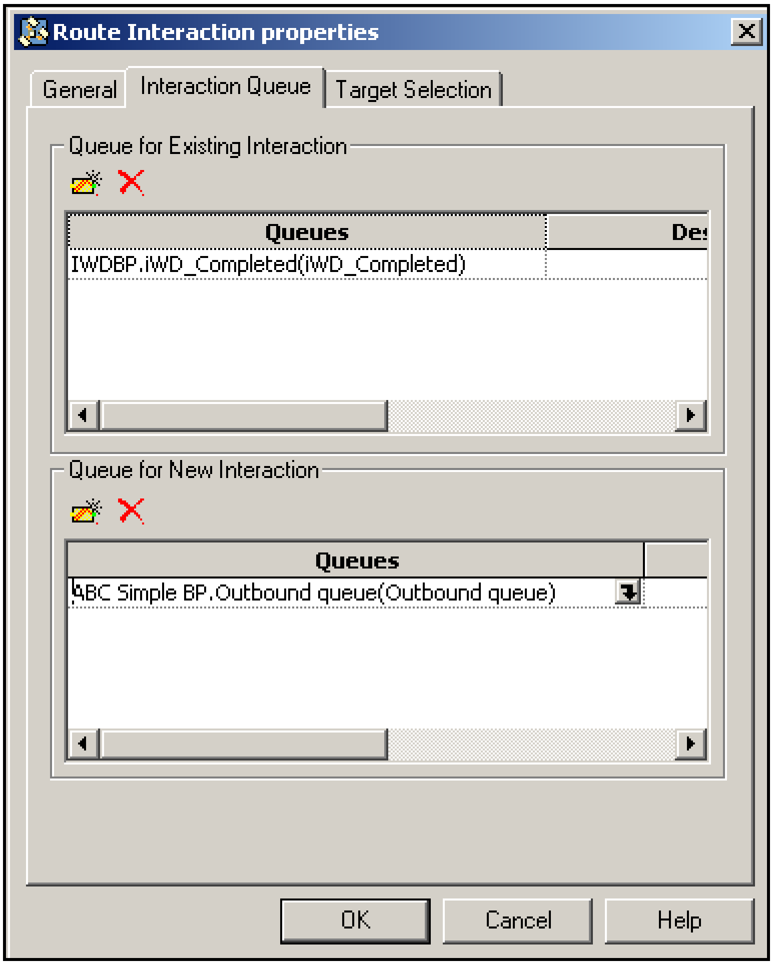
You might want to check all of the provided `Route Interaction` objects (by double-clicking on them) to see if they satisfy your business logic. Pay attention only to what statistics are used for `Skill` and `AgentGroup` routing.

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 properties on the Interaction Queue tabs of all the Route Interaction blocks in the strategy must be as shown in Figure 116.


Route Interaction properties
General
Interaction Queue
Target Selection
Queue for Existing Interaction
Queue for New Interaction
OK
Cancel
Help

Queue for Existing Interaction

Queues	Des
IWDBP.iWD_Completed(iWD_Completed)	

Queue for New Interaction

Queues	
ABC Simple BP.Outbound queue(Outbound queue)	

Figure 116: 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 and Genesys Rules System 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, which includes the Genesys Rules System

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 117.

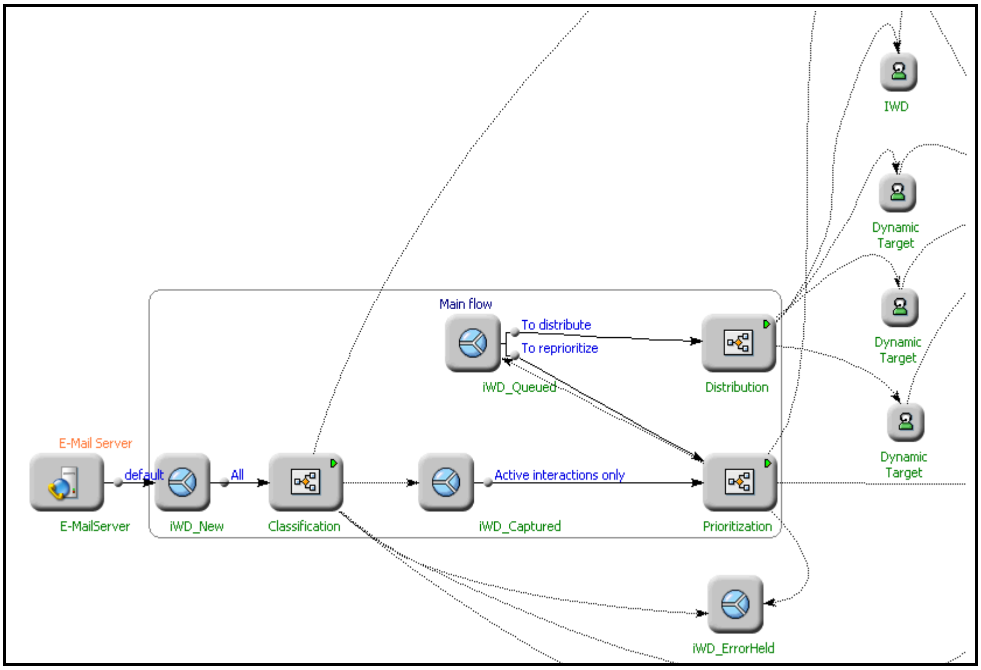
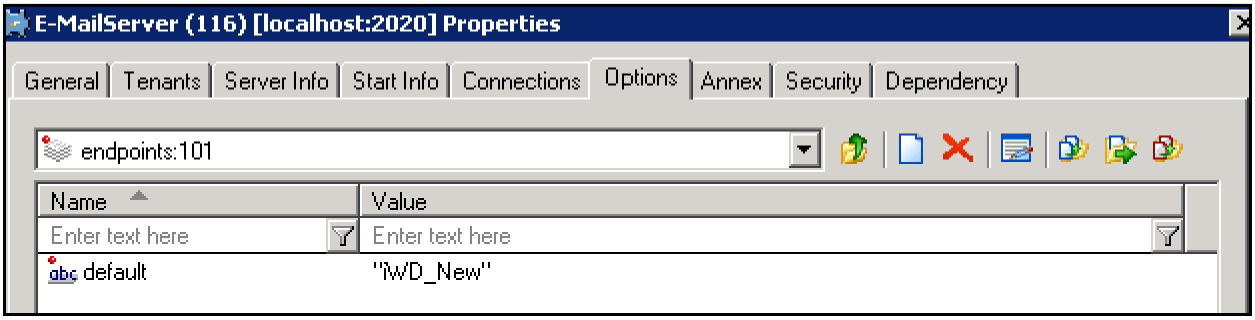


Figure 117: Main Flow of IWDBP

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 117. Next, refresh in IRD and E-mail Server will be added to the business process with the `iWD_New` queue.



endpoints:101

Name	Value
default	"iWD_New"

Figure 118: 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 must be increased by 5.

Preparation of Components

Genesys Configuration

To prepare the Genesys configuration:

- Add Agents into the `IWD` Agent Group.
- If it has not already been done, set up a connection between Interaction Server and both the Business Context Management Service `Application`, and the Genesys Rules System `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 350 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 (with ID also set to MCR). On the MCR tenant's Profile page, configure the URL for the Genesys Rules Authoring Tool.
- Under iWD tenant MCR create a new solution and name it MCR Solution with ID = MCR_SLT.
- 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 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.

Once your services are all created, you will need to do two more things:

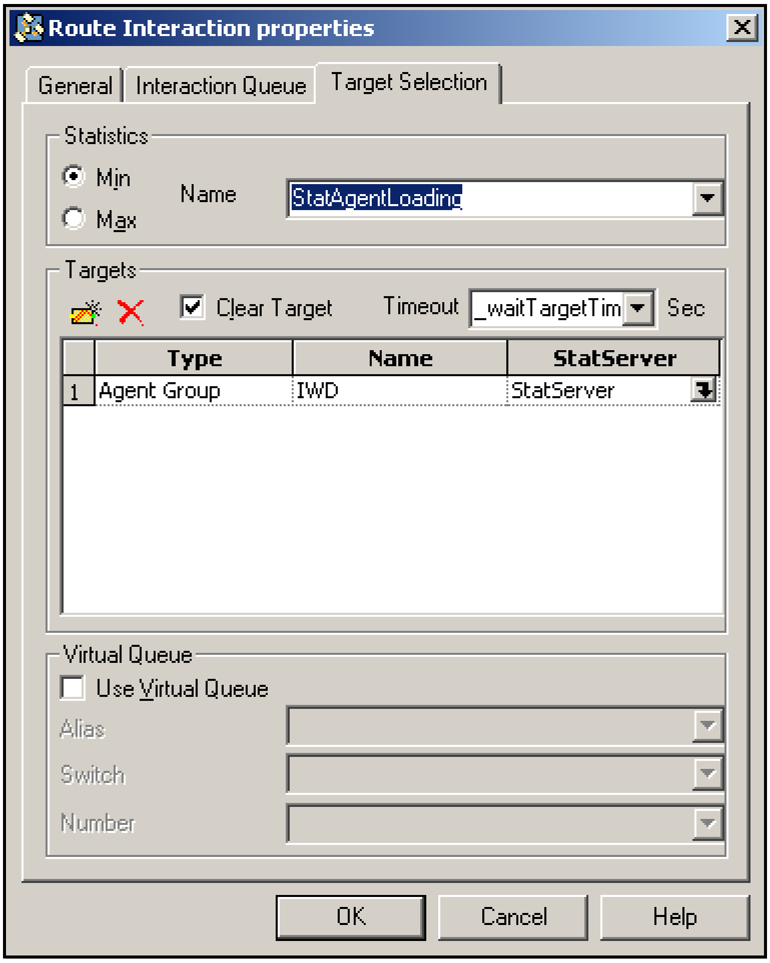
- Push the business structure changes to the Genesys Rules System.
- Deploy your iWD Solution to your application server.

You should see two notifications on the top of the iWD Manager screen informing you of these two tasks. You can select each of the hyperlinks to take you to the screen where the task needs to be performed.

IWDBP Preparation

To prepare the iWD business process:

- In IRD open `IWDBP`. Open the `Distribution` strategy. Double-click on the fourth `RouteInteraction` block (the one that is used to route to the IWD agent group). In the `Target Selection` tab, change the Genesys Stat Server application name and target according to your configuration. See Figure 119.



Statistics

● Min ○ Max Name: StatAgentLoading

Targets

☑ Clear Target Timeout: _waitTargetTim Sec

	Type	Name	StatServer
1	Agent Group	IWD	StatServer

Virtual Queue

□ Use Virtual Queue

Alias

Switch

Number

Figure 119: 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. Launch the Genesys Rules Development Tool (GRDT) and import the `iWD Standard Rules Template` project if it is not already there.

   If the template has not already been imported into GRDT, you can find the `iWD Standard Rules Template` in the directory where iWD Manager supporting files were installed.

2. Expand `Conditions` and add four new rule conditions, entering the `Language Expression` and `Rule Language Mapping` according to the information in Table 60.
3. Click `Save`.
4. Right-click the `iWD Standard Rules Template` and select `Publish`.

   It will be published to the `Genesys Rules System` rules repository.

Table 60: Conditions

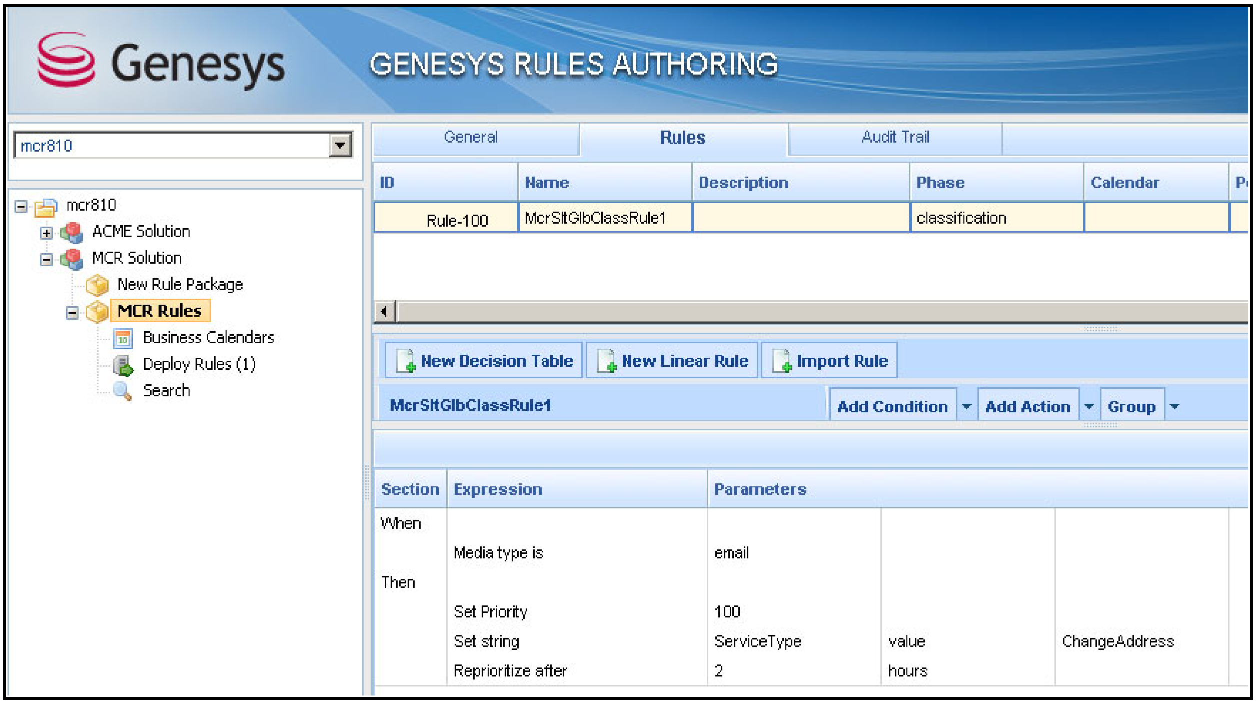
Language Expression	Rule Language Mapping
Reprioritization was not set up	`eval(!$data.containsKey("IWD_reprioritizeDateTime"))` **Note:** The `Standard Rules Template` contains a standard rule condition called `"Is first prioritization"` that does the same thing as `"Reprioritization was not set up"`, but `"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.
Reprioritization was set up and in the past	`eval( $data.containsKey("IWD_reprioritizeDateTime") && (getDTValue("IWD_reprioritizeDateTime", $data) < (getCurrentDT()) ))`
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 `KVPair "{k}" is "{v}"`. `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.
KVPair "{k}" is not "{v}"	`eval( !getStringValue('{k}', $data).equals('{v}'))` **Note:** You will also need to add two new parameters to the `Standard Rules Template`—`"k" and "v"`, which should both be of type `Input Value` with `Value Type=String`.

Create a Classification Rule

Create a classification rule (see Figure 120). For simplicity Global Rules (at the rule package level) are used in this example.

1. Log into iWD Manager and launch the Genesys Rules Authoring Tool.
2. Select your iWD tenant.
3. Expand your Solution (`MCR Solution`) and select `New Package`.
4. Give the rule package a `Package Name` (`mcr`) and a `Business Name` (for example, `MCR Rules`).
5. Select `Package Type iWD`.
6. Under `Templates`, select the `iWD Standard Rules Template` and click `Save`.
7. From the navigation tree, select your rule package (`MCR Rules`).
8. Click the `Rules` tab.

9. Select `New Linear Rule`.
10. Give the rule a `Name` and, as the rule phase, select `classification`.
11. Click `Add Condition`,
12. Select the `Media type is` rule condition.
13. In the condition, from the drop-down list of media types, select `email`.
14. Click `Add Action`
15. Select `Set Priority` and set the priority value to `100`.
16. Click `Add Action`.
17. Select `Set String` and set the `ServiceType` string equal to the value `ChangeAddress`.
18. Click `Add Action`.
19. Select `Reprioritize` and, for the reprioritization interval, enter `2 hours`.
20. Click `Save`.



ID	Name	Description	Phase	Calendar	P
Rule-100	McrSltGlbClassRule1		classification		

Section	Expression	Parameters		
When				
	Media type is	email		
Then				
	Set Priority	100		
	Set string	ServiceType	value	ChangeAddress
	Reprioritize after	2	hours	

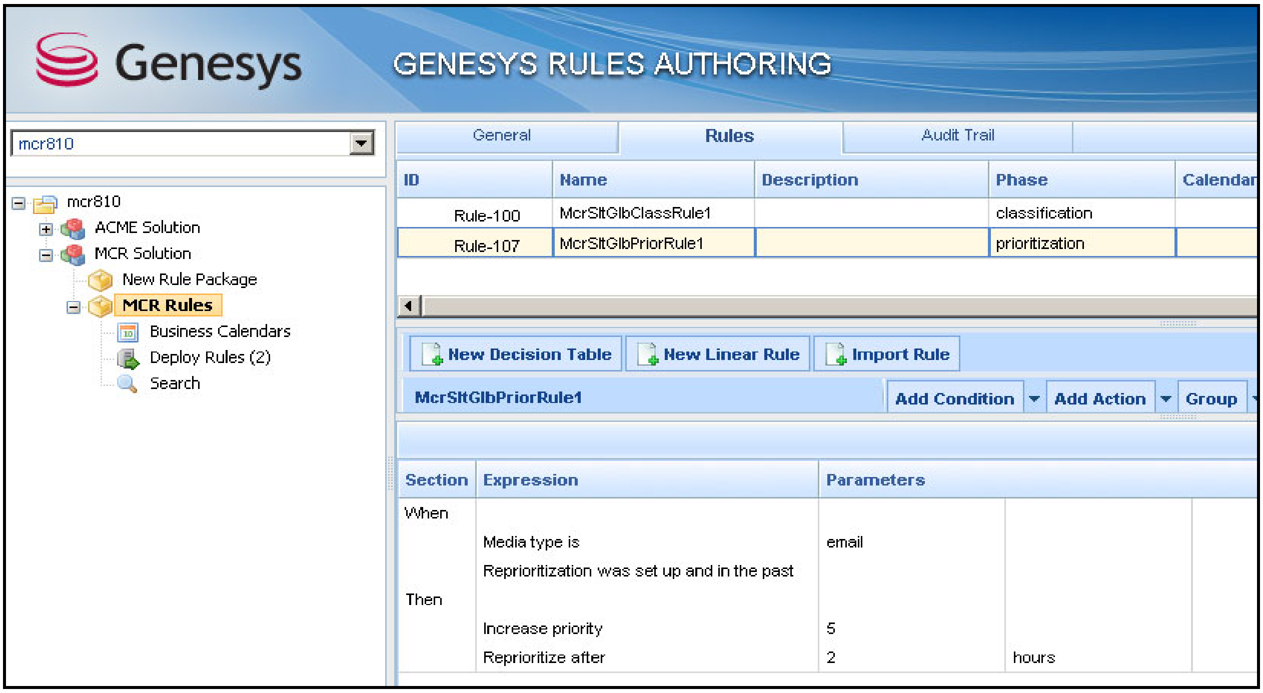
Figure 120: Create a Classification Rule

Create a Prioritization Rule

Create a prioritization rule (see Figure 121). For simplicity, Global Rules (at the rules package level) 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 level with the rule action `Assign iWD process`, to assign the interaction to an iWD Process.

1. Click `New Linear Rule`.
2. Select `prioritization` as the `Phase`.

3. Click Add Condition and select Media type is.
4. From the media-types drop-down list, select email.
5. Select Add Condition and select Reprioritization was set up and in the past.
6. Click Add Action, and select Increase priority.
7. For the amount the priority should be increased by, enter 5.
8. Click Add Action, and select Reprioritize after.
9. For the time at which to reprioritize the interaction, enter 2 hours (see Figure 121.
10. Click Save.



ID	Name	Description	Phase	Calendar
Rule-100	McrSltGlbClassRule1		classification	
Rule-107	McrSltGlbPriorRule1		prioritization	

Section	Expression	Parameters	
When			
	Media type is	email	
	Reprioritization was set up and in the past		
Then			
	Increase priority	5	
	Reprioritize after	2	hours

Figure 121: Create a Prioritization Rule

Deploy Modifications

After new rules are created the rule package has to be deployed before the rules will have an affect on your business process. Click on the Deploy Rules node in the Genesys Rules Authoring Tool (GRAT) navigation tree and select Deploy Now. You will receive a notification if the deployment was successful.

Note: There are two List Objects used in the IWDBP. One contains two lists and the other contains one list. All three lists must be properly configured for the IWDBP to work properly for your iWD Solution and your rule package. See “Configuration of List Objects” on page 314.

Notes on the iWD Business Process

The following are some important things to note about the iWD business process:

- Behavior of the iWD_Captured queue and Active interactions only 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 To reprioritize view—interactions will only be taken through this view with the key IWD_reprioritizeDateTime is in the past. This key-value pair is set up by a prioritization business rule. See “Create a Prioritization Rule” on page 355.

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 the most frequently. Interactions with Subject = Support will have lower priority and all other interactions will have the lowest priority and will be reprioritized less frequently.

Rules Creation

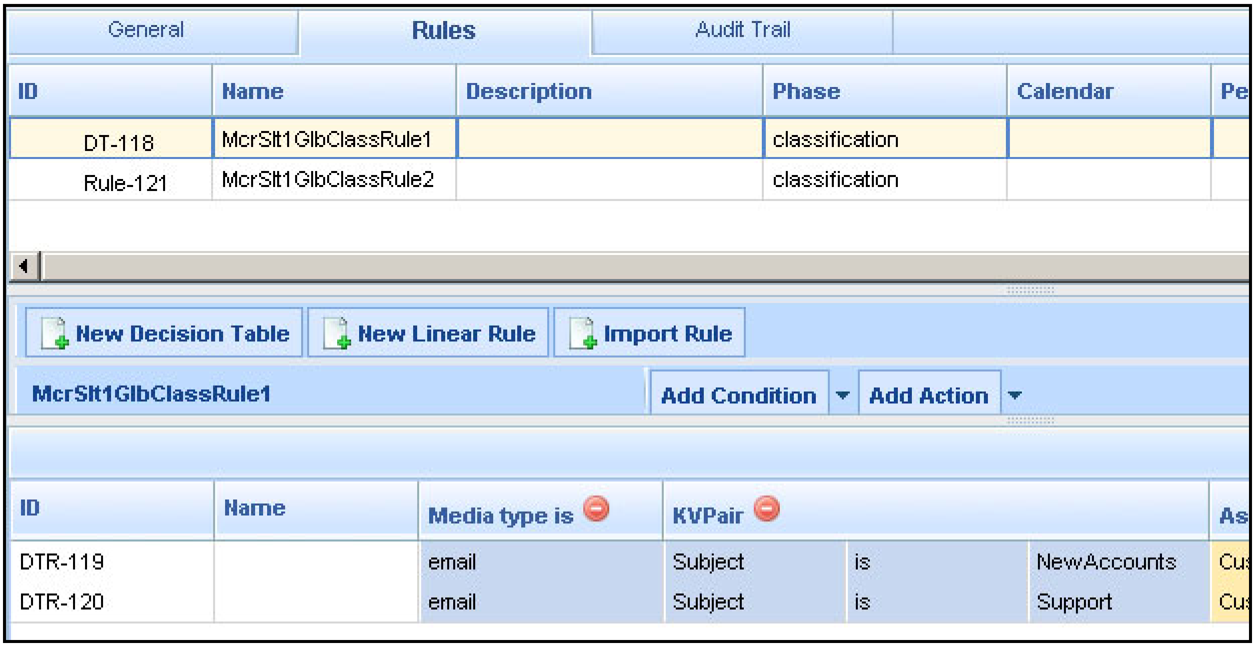
Create a Department and Processes in iWD

Create the Department and Processes in iWD Manager.

1. Select Departments & Processes.
2. Select your iWD tenant.
3. Select New Department.
4. Enter a name for the new department (in this example, Customer Service).
5. Click Save.
6. Expand the new Department, you named in Step 4.
7. Create the following three new Processes under the Customer Service Department: Sales, Support, and Other, clicking Save after each one is created.
8. Redeploy your iWD Solution
9. Push the business structure changes to the Genesys Rules System.

Create Global Classification Rules

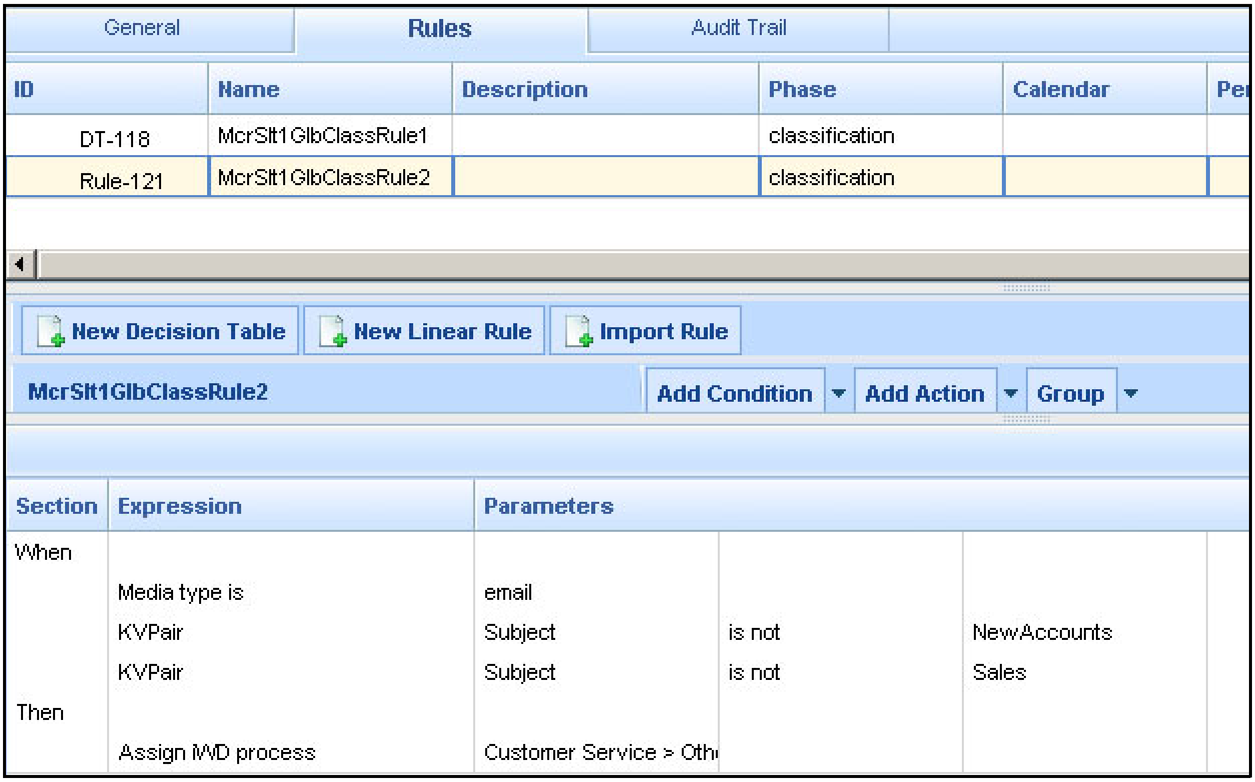
Create new Global classification rules as shown in Figure 122 and Figure 123. Remember to save each rule. Figure 122 shows an example of a Decision Table rule.



ID	Name	Description	Phase	Calendar	Pe
DT-118	McrSlt1GlbClassRule1		classification		
Rule-121	McrSlt1GlbClassRule2		classification		

ID	Name	Media type is	KVPair			As
DTR-119		email	Subject	is	NewAccounts	Cu
DTR-120		email	Subject	is	Support	Cu

Figure 122: Classification Rules for Example 2—Rule 1



ID	Name	Description	Phase	Calendar	Pe
DT-118	McrSlt1GlbClassRule1		classification		
Rule-121	McrSlt1GlbClassRule2		classification		

Section	Expression	Parameters		
When				
	Media type is	email		
	KVPair	Subject	is not	NewAccounts
	KVPair	Subject	is not	Sales
Then				
	Assign iWD process	Customer Service > Oth		

Figure 123: Classification Rules for Example 2—Rule 2

Create Classification Rules for each Process

Create a new classification rule for each process (use Figure 124 as an example).

For the `Sales` process:

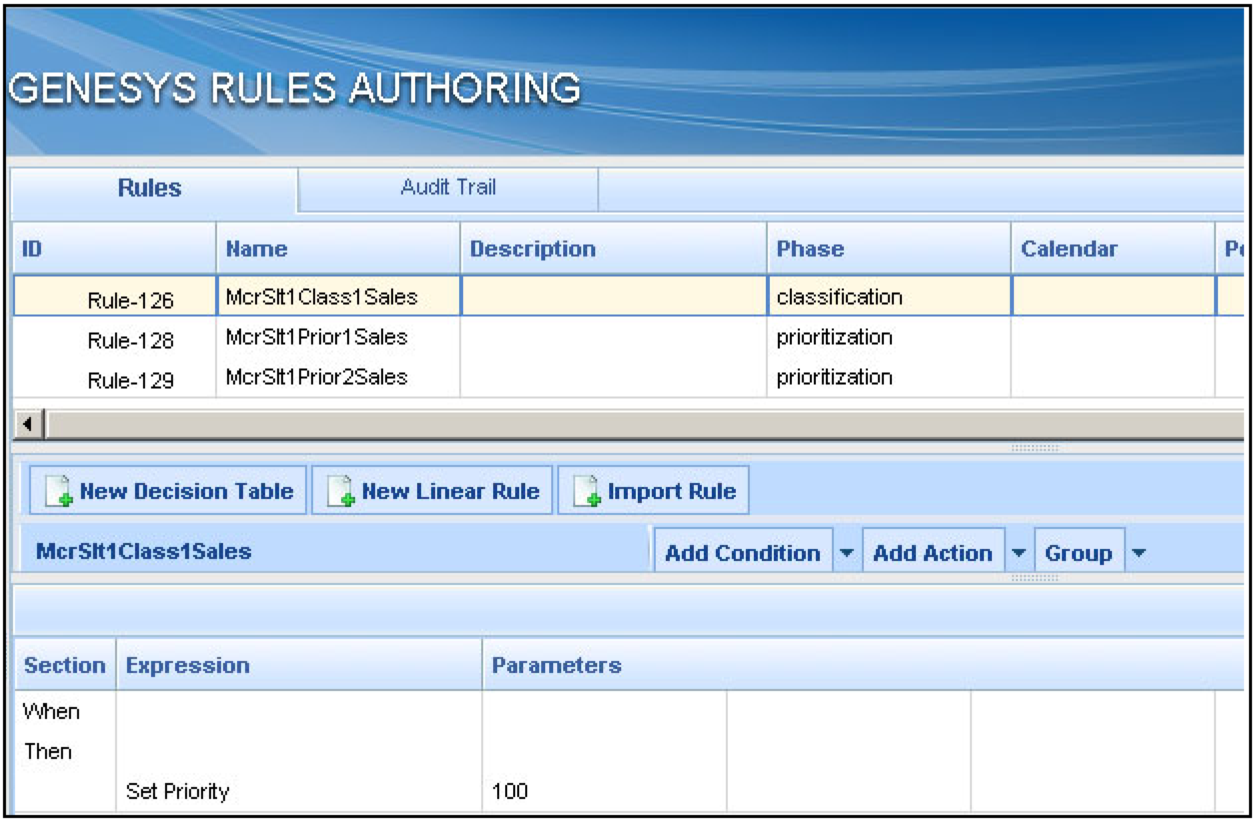
Action `Set Priority 100`

For the `Support` process:

Action `Set Priority 50`

For the `Others` process:

Action `Set Priority 10`



ID	Name	Description	Phase	Calendar	P
Rule-126	McrSlt1Class1Sales		classification		
Rule-128	McrSlt1Prior1Sales		prioritization		
Rule-129	McrSlt1Prior2Sales		prioritization		

McrSlt1Class1Sales

Section	Expression	Parameters
When		
Then		
	Set Priority	100

Figure 124: Example Classification Rule for a Process

Create Prioritization Rules for the Processes

Create prioritization rules for your processes. See Figure 125 for an example.

For the `Sales` process, create the rules outlined in Table 61.

Table 61: Sales Process Prioritization Rules

Rule Name	Conditions	Actions
McrSlt1Prior1Sales	If Reprioritization was not set up	Reprioritize after 1 hour
McrSlt1Prior2Sales	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 62.

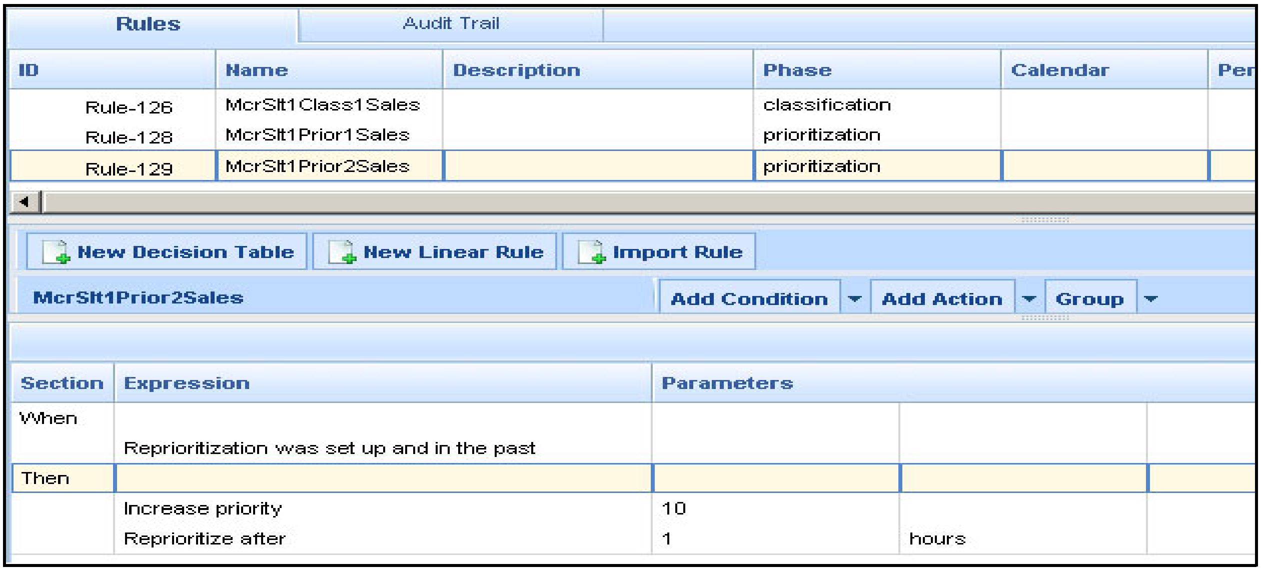
Table 62: Support Process Prioritization Rules

Rule Name	Conditions	Actions
McrSlt1Prior1Support	If Reprioritization was not set up	Reprioritize after 2 hours
McrSlt1Prior2Support	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 63.

Table 63: 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



ID	Name	Description	Phase	Calendar	Per
Rule-126	McrSlt1Class1Sales		classification		
Rule-128	McrSlt1Prior1Sales		prioritization		
Rule-129	McrSlt1Prior2Sales		prioritization		

Section	Expression	Parameters		
When				
	Reprioritization was set up and in the past			
Then				
	Increase priority	10		
	Reprioritize after	1	hours	

Figure 125: Prioritization Rules for the Sales Process

Deploy Changes Deploy your rule package. Refer to page 356.

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:

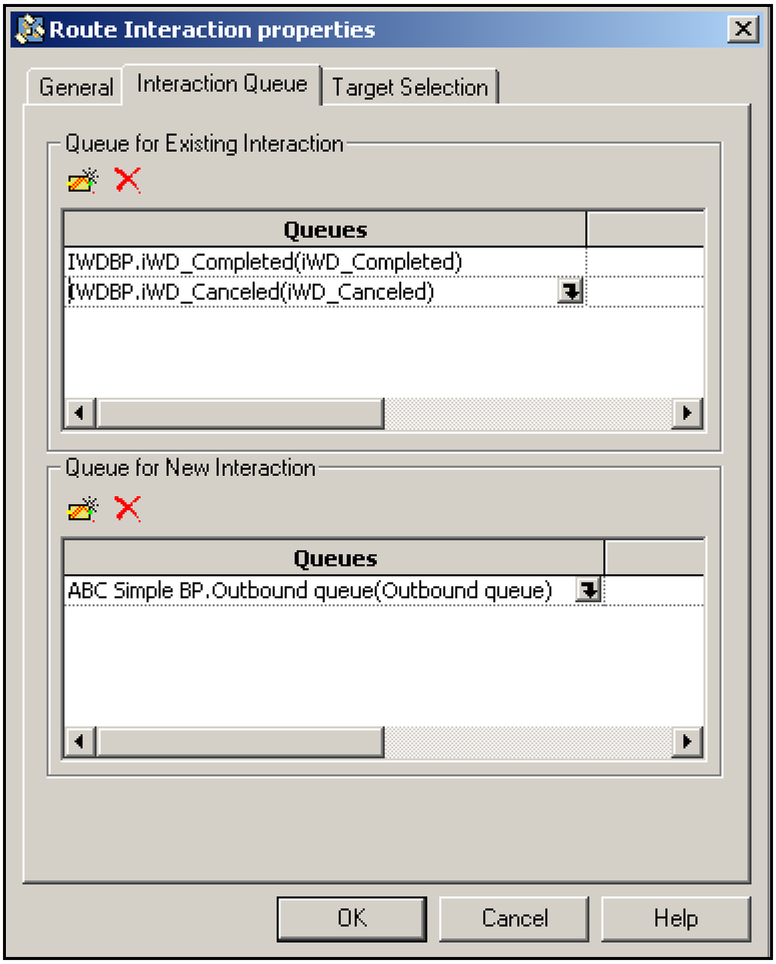
- The interaction is submitted by E-mail server and is placed into the `iWD_New` queue.
- The interaction is processed by the `Classification` strategy. As a result, the classification business rules will be applied to the interaction. In this example, interaction will be assigned to one of the iWD processes 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 126.



Queue for Existing Interaction

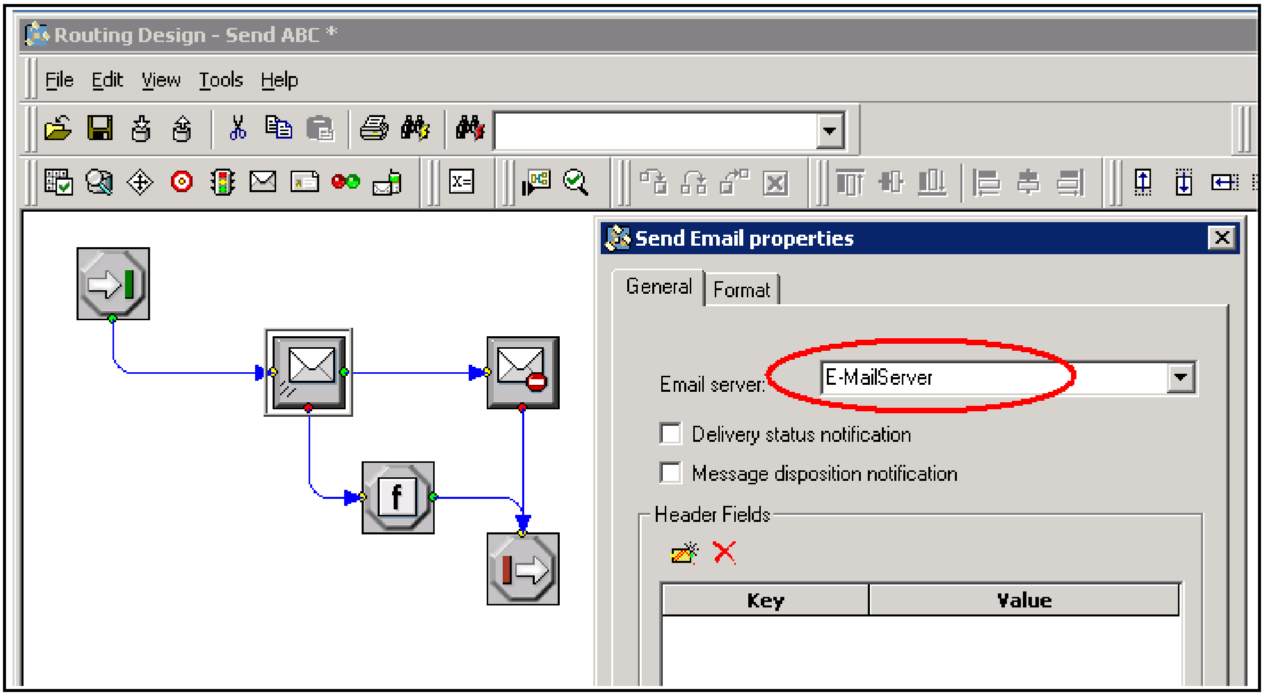
Queues
IWDBP.iWD_Completed(iWD_Completed)
IWDBP.iWD_Canceled(iWD_Canceled)

Queue for New Interaction

Queues
ABC Simple BP.Outbound queue(Outbound queue)

Figure 126: 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 127).



Email server: E-MailServer

☐ Delivery status notification

☐ Message disposition notification

Header Fields

Key	Value

Figure 127: Select E-Mail Server

After these modifications the iWD business process should look approximately like Figure 128.

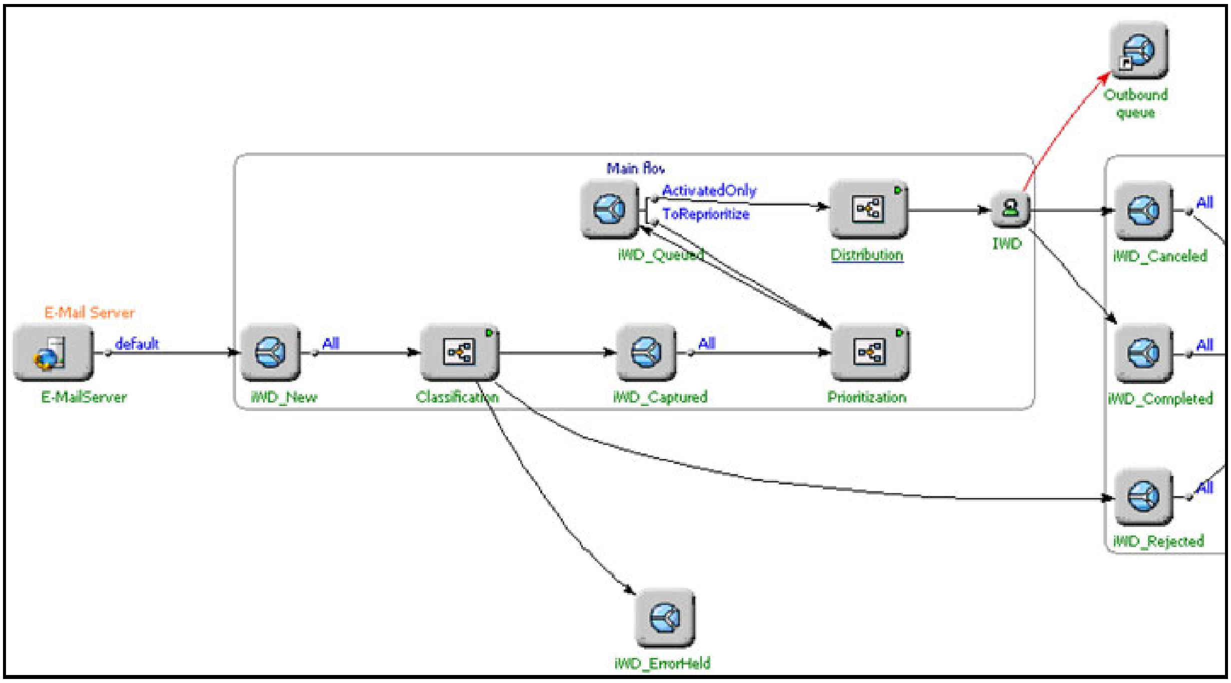


Figure 128: iWD Business Process after Modifications

Save all modifications and run all participating strategies.

Figure 129 shows the `ABC Simple` business process.

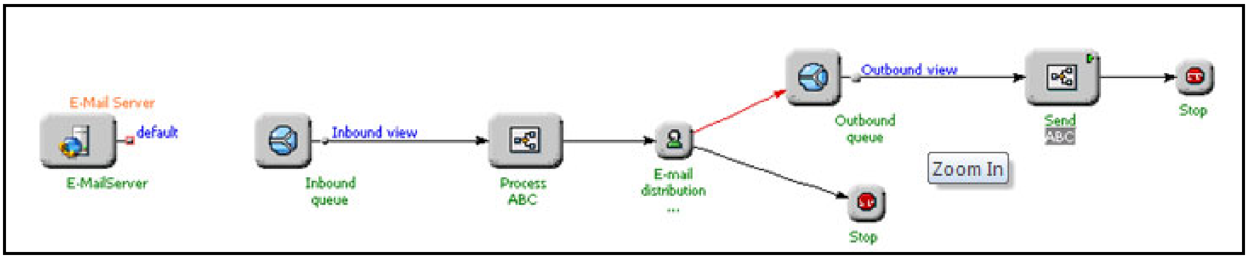


Figure 129: ABC Simple Business Process

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.

Genesys

Appendix

E iWD Rules and Existing Business Processes

This Appendix explains how to use iWD business rules functionality with existing business processes. It explains the modifications that are required to use iWD business rules within existing business processes.

This Appendix contains the following sections:

Using iWD Rules in Existing Business Processes

The requirements, assumptions, and examples in this section provide information about how to use iWD Rules in existing Business Processes.

Requirements:

- A Genesys E-mail solution or any other Genesys eServices 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, including the Genesys Rules System, 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 `Iwd_Esp_List` and `Iwd_Package_List` List Objects accordingly (see “Configuration of List Objects” on page 314).

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 135 on page 372 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 the classification rules in your rule package accordingly (add the 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 the eServices Interaction Workflow installation.

ABC Simple BP

The example uses the ABC Simple BP business process that comes with the eServices Interaction Workflow installation. See Figure 130.

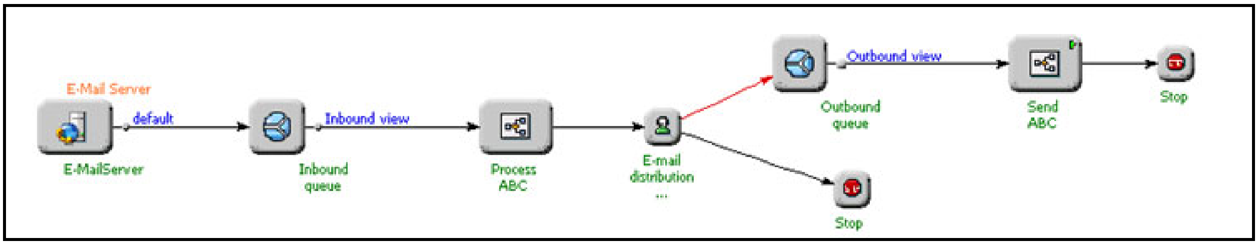


Figure 130: ABC Simple BP

Example

This is a simple example that shows how a business process can 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 the highest priority and will be reprioritized more frequently. Interactions with `Subject = Support` will have lower priority and all other interactions will have the lowest priority and will have the least frequent reprioritization.

Modify the Existing Business Process

Modify the existing business process. Add one more strategy, which will invoke the iWD Business Content Management Service (BCMS) to prepare the interaction user data for rule evaluation, followed by the Genesys Rules Engine 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 131 shows the `ABC IWD Simple BP` business process.

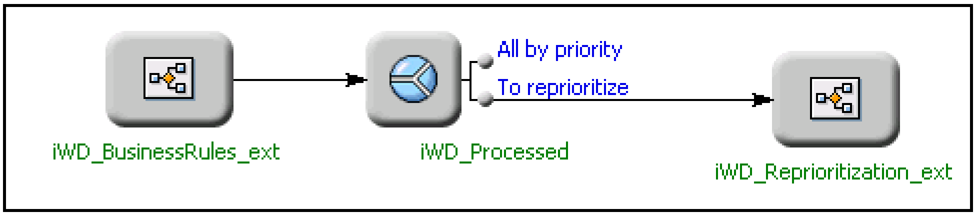
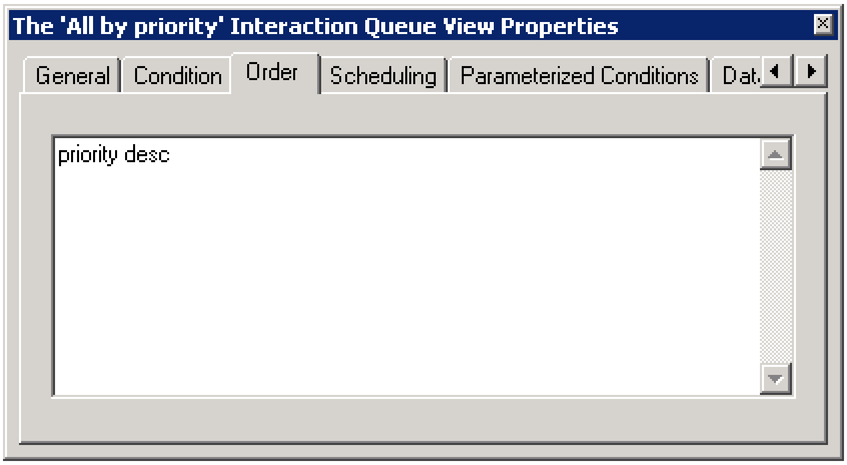


Figure 131: ABC IWD Simple BP

Note: The `iWD_BusinessRules_ext` and `iWD_Reprioritization_ext` routing strategies (see Figure 131) use the two List Objects described in “Configuration of List Objects” on page 314.

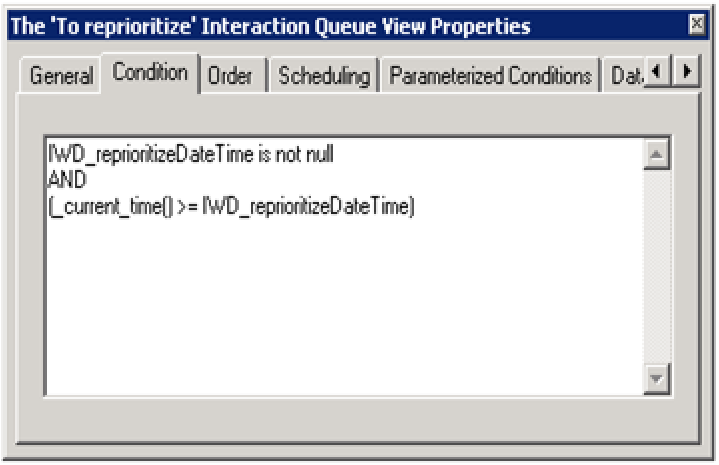
In this business process, the property of the `All by priority` view is configured as shown in Figure 131 (the `Conditions` tab is empty). Thus, interactions from this queue and through this view will be taken by priority. The interaction with the highest priority will be taken first.



```
priority desc
```

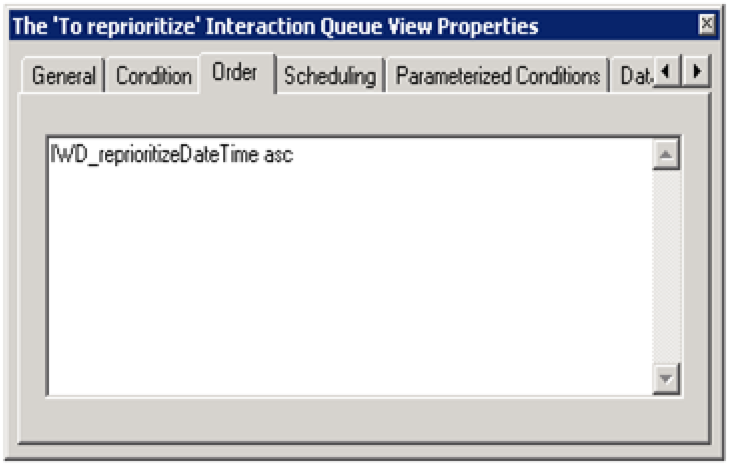
Figure 132: All by Priority Interaction Queue View Properties

Properties of the `To reprioritize` view are configured as in Figure 133 and Figure 134. 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.



```
IWD_reprioritizeDateTime is not null
AND
(_current_time() >= IWD_reprioritizeDateTime)
```

Figure 133: 'To Reprioritize' Properties—Condition Tab



IWD_reprioritizeDateTime asc

Figure 134: ‘To Reprioritize’ Properties—Order Tab

Figure 135 shows the iWD Business Rules Ext strategy.

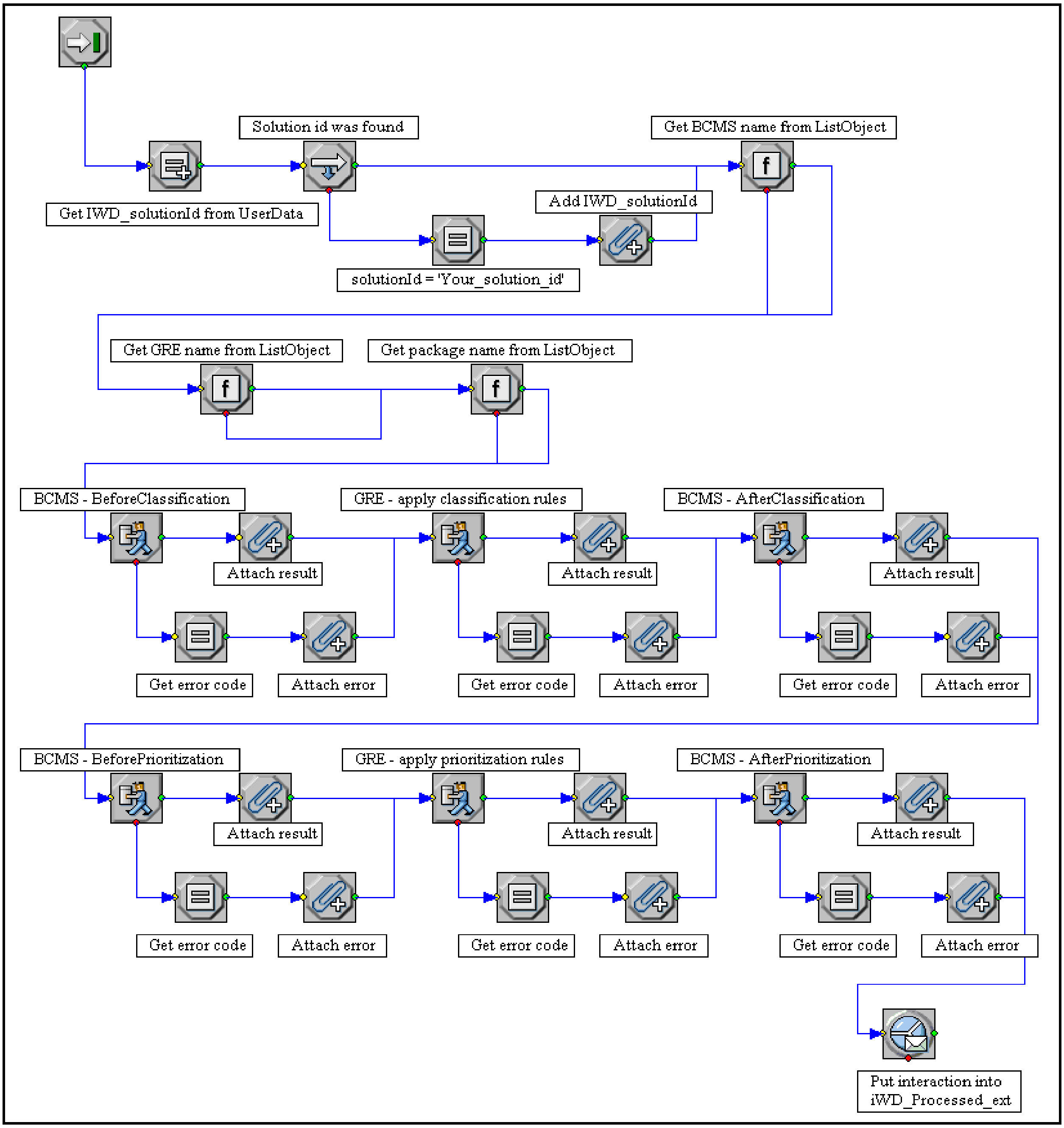

Solution id was found
Get BCMS name from ListObject
Get IWD_solutionId from UserData
Add IWD_solutionId
solutionId = 'Your_solution_id'
Get GRE name from ListObject
Get package name from ListObject
BCMS - BeforeClassification
GRE - apply classification rules
BCMS - AfterClassification
Attach result
Attach result
Attach result
Get error code
Attach error
Get error code
Attach error
Get error code
Attach error
BCMS - BeforePrioritization
GRE - apply prioritization rules
BCMS - AfterPrioritization
Attach result
Attach result
Attach result
Get error code
Attach error
Get error code
Attach error
Get error code
Attach error
Put interaction into
iWD_Processed_ext

Figure 135: iWD Business Rules Ext Strategy

Figure 136 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.

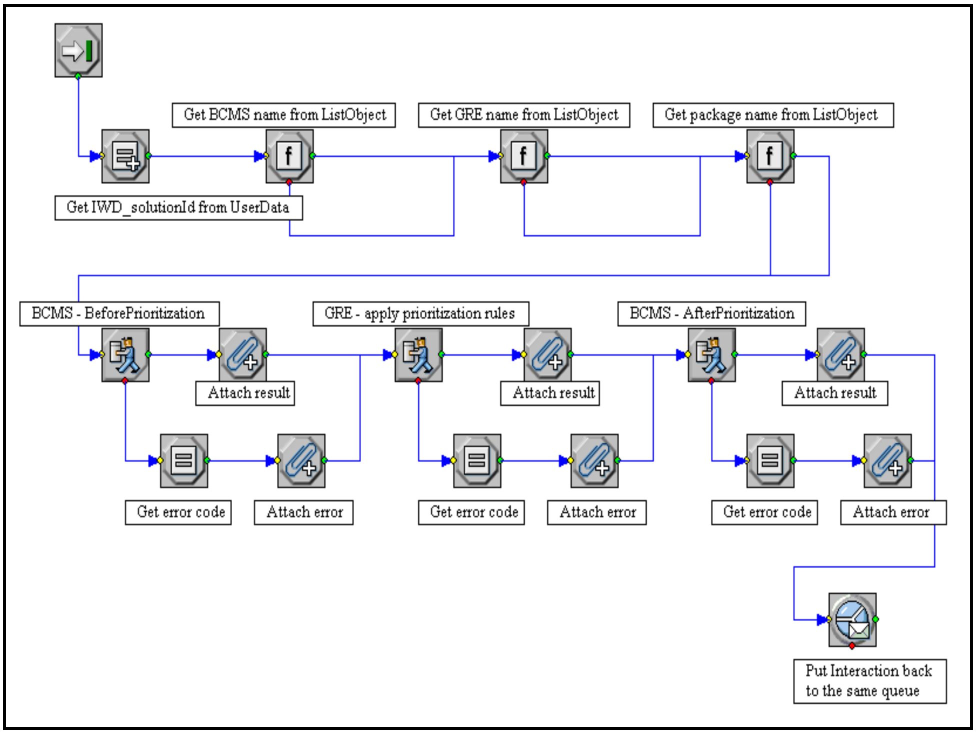

Get BCMS name from ListObject
Get GRE name from ListObject
Get package name from ListObject
Get IWD_solutionId from UserData
BCMS - BeforePrioritization
GRE - apply prioritization rules
BCMS - AfterPrioritization
Attach result
Attach result
Attach result
Get error code
Attach error
Get error code
Attach error
Get error code
Attach error
Put Interaction back to the same queue

Figure 136: 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 137 shows how the ABC Simple BP will look at this point.

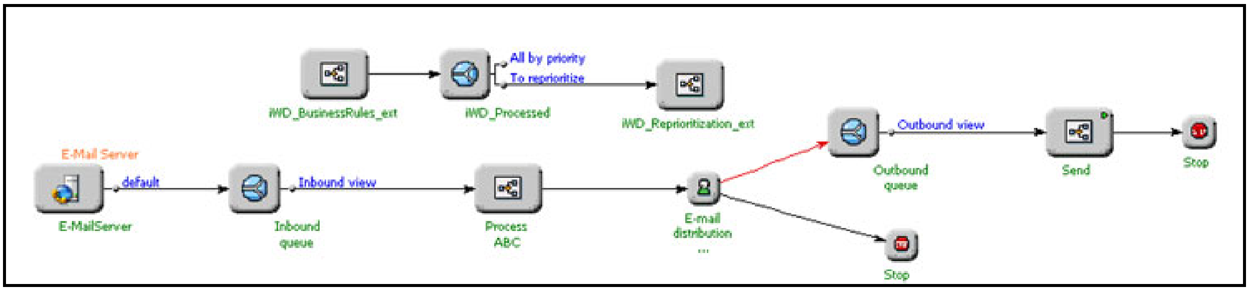


Figure 137: ABC Simple BP with Queue and Strategies Added

Insert the added group between Inbound queue and Process ABC. Figure 138 shows how the ABC Simple BP business process will look at this point.

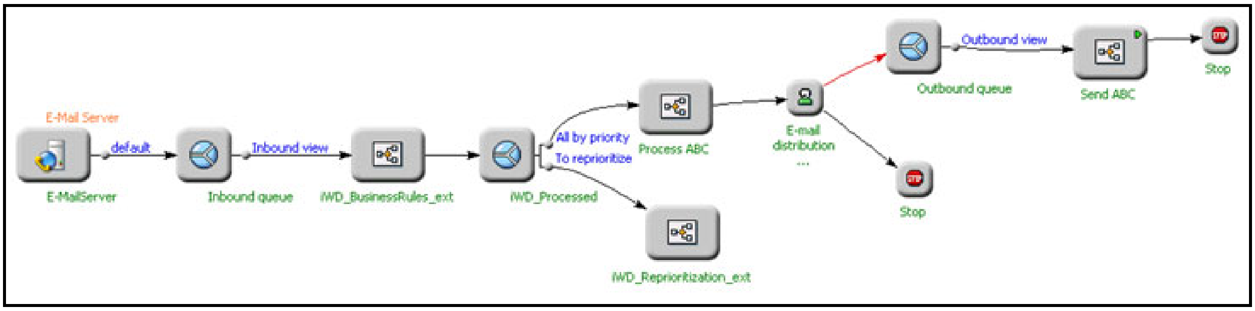


Figure 138: ABC Simple BP with Group Added

The existing business process is now updated. Next, create business rules in iWD.

Rules creation

Open Genesys Rules Development Tool and modify the Standard Rules Template as described in “Modify the Standard Rules Template” on page 353. 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 352.

Create a department and processes as described in “Create a Department and Processes in iWD” on page 358.

In Genesys Rules Authoring Tool, create a rule package and add the iWD Standard Rules Template to the rule package. Create new (package-level) classification rules as described in “Create Global Classification Rules” on page 358 (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 361.

Deploy your iWD rule package and your iWD Solution.

Path of E-mail Interactions

The interaction will pass through the following are the steps:

1. The interaction is submitted by the Genesys E-mail server and is placed into the Inbound queue.
2. The interaction is processed by the IWD_BusinessRules_Ext strategy:
    a. The BMCS service is invoked by using the Method BeforeClassification, to ensure the integrity of the interaction user data prior to the Genesys Rules Engine being invoked.
    b. The Genesys Rules Engine is invoked:

        - The Genesys Rules Engine is called and the global (package-level) 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 BMCS service is invoked with the Method AfterClassification, to ensure the integrity of the interaction user data after the Genesys Rules Engine was invoked.

    c. The BMCS service is invoked with the Method BeforePrioritization, to ensure the integrity of the interaction user data prior to the Genesys Rules Engine being invoked.

    d. The Genesys Rules Engine is invoked again, to evaluate the prioritization rules.

        - 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).
        - Finally, the BMCS service is invoked with the Method AfterPrioritization, to ensure the integrity of the interaction user data after the Genesys Rules Engine was invoked.

3. The IWD_BusinessRules_Ext strategy is completed and the interaction is placed into the iWD_Processed queue.

4. 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.
- Release Notes and Product Advisories for this product, which are available on the Genesys Customer Care website at `http://genesys.com/customer-care`.

eServices

- *eServices 8.1 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 Integrated Capture Points.
- *eServices 8.1 User's Guide,* which provides overall information and recommendations on the use and operation of Genesys eServices, and the use and operation of the Integrated Capture Points.
- *eServices 8.1 Reference Manual*, which provides information about the configuration options for each eServices component, including options specific to the Integrated Capture Points.
- *eServices 8.1 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 8.1 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,* available on the Genesys Documentation website, provides a comprehensive list of the Genesys and computer-telephony integration (CTI) terminology and acronyms used in this document.
- *Genesys Migration Guide*, available on the Genesys Documentation website and which ships on the Genesys Documentation Library DVD, provides documented migration strategies for Genesys product releases. Contact Genesys Customer Care for more information.

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

- Genesys Supported Operating Environment Reference Guide
- 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 Documentation website (docs.genesys.com).

Genesys product documentation is available on the:

- Genesys Customer Care website at `http://genesys.com/customer-care`.
- Genesys Documentation site at `http://docs.genesys.com/`.
- Genesys Documentation Library DVD, which you can order by e-mail from Genesys Order Management at `orderman@genesys.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 64 describes and illustrates the type conventions that are used in this document.

Table 64: Type Styles

Type Style	Used For	Examples
Italic	- Document titles - Emphasis - Definitions of (or first references to) unfamiliar terms - Mathematical variables Also used to indicate placeholder text within code samples or commands, in the special case where angle brackets are a required part of the syntax (see the note about angle brackets on page 380).	Please consult the *Genesys Migration Guide* for more information. Do *not* use this value for this option. A *customary and usual* practice is one that is widely accepted and used within a particular industry or profession. The formula, $x + 1 = 7$ where x stands for . . .
Monospace font (Looks like `teletype` or `typewriter text`)	All programming identifiers and GUI elements. This convention includes: - The *names* of directories, files, folders, configuration objects, paths, scripts, dialog boxes, options, fields, text and list boxes, operational modes, all buttons (including radio buttons), check boxes, commands, tabs, CTI events, and error messages. - The values of options. - Logical arguments and command syntax. - Code samples. Also used for any text that users must manually enter during a configuration or installation procedure, or on a command line.	Select the `Show variables on screen` check box. In the `Operand` text box, enter your formula. Click `OK` to exit the `Properties` dialog box. T-Server distributes the error messages in `EventError` events. If you select `true` for the `inbound-bsns-calls` option, all established inbound calls on a local agent are considered business calls. Enter `exit` on the command line.
Square brackets ([ ])	A particular parameter or value that is optional within a logical argument, a command, or some programming syntax. That is, the presence of the parameter or value is not required to resolve the argument, command, or block of code. The user decides whether to include this optional information.	`smcp_server -host [/flags]`
Angle brackets (< >)	A placeholder for a value that the user must specify. This might be a DN or a port number specific to your enterprise. **Note:** In some cases, angle brackets are required characters in code syntax (for example, in XML schemas). In these cases, italic text is used for placeholder values.	`smcp_server -host <confighost>`

Index