INTELLIGENT WORKLOAD DISTRIBUTION

High-Availability Reference Guide

April 2009
iWD 7.6.1

Copyright

Disclaimer

Feedback

Genesys endeavours to provide accurate and useful documentation for all Genesys products. To achieve this goal, the documentation group welcomes your comments and suggestions regarding any aspect of Genesys user documentation. Send your comments by e-mail to: techpubs.webadmin@genesyslab.com.

Trademarks and Registered Trademarks

Revision History

Version	Issued	Description
1.4.14	23.09.2008	Final Release for GTL release 1.4.14
1.5.6	10.11.2008	Final Release for GTL release 1.5.6
1.6.6	09.04.2009	Final Release for GTL release 1.6.6
7.6.1.1	16.04.2009	Rebranding to iWD 7.6.1
7.6.1.2	20.05.2009	Final Release for iWD 7.6.1

TABLE OF CONTENTS

Preface

Welcome to the *High-Availability Reference Guide*. This document provides a detailed description of the challenges the enterprises face and that are related to system availability and the solutions that intelligent Workload Distribution (iWD) offers towards addressing them.

This preface provides an overview of this guide, identifies the primary audience, introduces document conventions, and lists related reference information:

- Intended Audience
- Recommended Reading
- Chapter Summaries
- Document Conventions
- Related Resources

Intended Audience

This guide is intended for architects and consultants who are implementing iWD. This guide assumes the reader understands the iWD application and functionality, as well as software-architecture principles.

Recommended Reading

The reader is strongly encouraged to review the *iWD Overview Guide* as it introduces the main iWD concepts and provides a summary of the application functionality with iWD. In addition, the reader should review the *iWD Deployment Guide* as it contains further insights into the configuration and management of iWD.

Chapter Summaries

In addition to this preface, this guide contains the following chapters:

- Chapter 1: High Availability Challenges
- Chapter 2: High Availability Solutions

Document Conventions

This document uses the following stylistic and typographical conventions, which serve to identify specific types of information:

Type Styles

Italic

In this document, italic text denotes emphasis, document titles, definitions of (or first references to) unfamiliar terms, and mathematical variables. For example:

- Please consult the *intelligent Workload Distribution Manager User Guide* for more information.
- *Do not use* this value for this option.
- The formula, x +1 = 7 where x stands for . . .

Monospace Font

A monospace font, which resembles `teletype` or `typewriter` text, is used for all programming identifiers and graphical user interface (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; and code samples. For example:

- Select the `Default` check box.
- Click the `Edit` button.
- In the `Properties` dialog box, enter the value for the host server in your environment.
- Click `OK` to exit the `Properties` dialog box.

Monospace is also used for any text that users must manually enter during a configuration or installation procedure, or on a command line. For example: Enter `exit` on the command line.

Screen Captures in This Document

Screen captures of the product UI, as used in this document, can sometimes contain a minor spelling, capitalization, or grammatical error. The text that accompanies and explains each screen capture corrects such errors, *except* when such a correction might prevent you from installing, configuring, or successfully using the product. For example, if the name of an option contains a usage error, the name is presented exactly as it appears in the product GUI, without correction in any accompanying text.

Square Brackets

Square brackets indicate that a specific parameter or value is optional within a logical argument, a command, or some programming syntax. That is, the parameter's or value's presence is not required to resolve the argument, command, or block of code. You decide (or the user decides) whether to include this optional information. For example: `smcp_server -host [/flags]`

Angle Brackets

Angle brackets indicate a placeholder for a value that you (or the user) must specify. This might be a DN or port number that is specific to your enterprise. For example:
`smcp_server -host <configihost>`

Related Resources

Consult these additional resources as necessary:

- *iWD Overview*
- *iWD Deployment Guide*

Chapter 1 – High Availability Challenges

This chapter explains the challenges faced in maintaining a resilient operating environment and the contributors to system downtime. The information in this chapter includes the following topics:

For each topic, a discussion on how iWD addresses these challenges is included below.

Changing Environments

To stay competitive, businesses must constantly improve the way in which business processes and resources are managed. As businesses grow, they need to support new processes, expand operations, and optimize resource utilization.

Upgrades to software applications and environments can result in considerably lengthy process, and may include the temporary shut-down of services while configuration and/or software is undergoing an upgrade. This can require the careful synchronization of applied changes to the environment across all others; and during this synchronization, the entire system might become unavailable for a prolonged period.

intelligent Workload Distribution addresses these issues via:

- Incremental distribution of configuration: iWD avoids lengthy downtime and synchronization issues through incremental distribution of changes to all iWD services at the same time.
- Service redundancy: Enables upgrade of software services with a minimum impact on overall system's availability. Most iWD services support service backups in the service configuration, allowing system administrators the option of applying changes to the backup service, first while the primary service continues to process tasks. Upon completion of the backup-service upgrade, the primary service can be stopped, which triggers its backup service to fill in while the upgrades are applied to the primary service, without an interruption in system availability.

Changing an environment also exposes a significant risk of introducing human errors, which is the topic of the next section.

Human Error

A common source of system downtime is human error. While these errors can never be completely avoided, there are various ways in which to minimize their likelihood. iWD incorporates a number of preventive and corrective measures that are designed to:

- Avoid errors that occur when an operator enters incorrect data.
- Avoid errors that occur when the user performs an incorrect action.
- When required, identify and resolve human errors quickly.

The following table lists different problems that relate to human error and the measures that iWD takes to address them:

Problem	Measures
Inconsistency in the specification of the same entity in two different pieces of a system	✓ Centralized configuration ✓ "Do not repeat yourself" principle ✓ Incremental distribution of configuration
Unauthorized or unskilled changes to system that result in critical reductions of availability	✓ Validation and integrity rules ✓ User-access control
It's often cumbersome and time-consuming to identify what causes a system to malfunction	✓ Configuration auditing and versioning

Hardware Failure

Hardware failures include disk and network failures, processor-unit and memory failures, loss of power supply, and internal system cooling.

As with other systems, intelligent Workload Distribution software solutions operate on hardware platforms that are connected by a WAN/LAN and can be subject to outages that are based on the previously listed causes.

Various computer-operating platforms offer different platform-specific solutions for availability. This includes both traditional duplication of hardware systems with fully replicated sets of components, preventing single points of failure, and hot-swappable components, allowing repair while the computer is online. Clustering, where groups of computers act as a single system from an external viewpoint, provides the highest end of availability.

These hardware solutions also vary from supplier to supplier. Following its commitment to hardware-platform independence, Genesys has implemented fault-tolerance capabilities at the application level within iWD. This is via transaction and communication management and service redundancy, described in the next chapter.

Software Failure

Software failures can result through an exception in the operating system, middleware, or application itself. An exception is an interruption in the normal flow of a program that is caused by an internal defect.

The most effective measure against software failures is detecting and fixing defects early in the iWD product lifecycle. All iWD solutions are submitted to extensive review and testing at each step of their development. All services undergo a wide range of functional, integration, and stress tests.

Additionally, intelligent Workload Distribution provides an integrated service-management and monitoring layer, providing detection and insight into a failure situation in a timely and coordinated fashion.

Communication Failure

Communication failure is specific to type of hardware (like physical damage to a network router or link device) or software failure (like a flaw in the network TCP/IP routing logic).

In distributed systems, services must communicate continuously with each other and, in some cases, with external resources. Whenever a solution component cannot communicate with a required resource, the solution may not be able to perform its required function.

While most designs will provide duplicate communication paths to avoid single points of communication failure, iWD implements a number of measures to handle and tolerate communication failures gracefully at the iWD application level:

- Queued Communication
- Connection Management

Chapter 2 – High Availability Solutions

This chapter describes different types of solutions that iWD incorporates to address high-availability challenges. These solutions focus on five specific software-system aspects, each critical for system's availability:

- Configuration: An intuitive approach to managing configuration changes, in a way that minimizes system interruption
- Management & Monitoring: Provide facilities to react quickly to hardware, software, or communication failure
- Transactions: Ensure consistency of data in case of hardware, software, or communication failure
- Communications: Ensure that all intended data is always delivered to other party, whether internal or external
- Redundancy: Maintains overall system availability, in case of hardware or communication failure

Configuration

The focus on configuration over coding in iWD allows changes to business and technical configurations to be made with ease, and without requiring a significant amount of time. Where changes are required, it is important that these are completed in a centralized and straightforward approach, so as to avoid duplication, guard against invalid input by users, and audit all changes to the system.

This section describes key functionality within iWD that provides safeguards against any system outage that might be caused by incorrect configuration. They are the following:

- Centralized Configuration
- Incremental Distribution
- "Do not repeat yourself" principle
- Validation and Integrity Checks
- User Access Control
- Change Control

Centralized Configuration

The centralized configuration management in iWD provides a single and consistent interface for business and technical configurations, while reducing the time that is required to implement configuration changes. All changes are through a role-based, intuitive thin-client UI, of which a sample is shown in the following image:

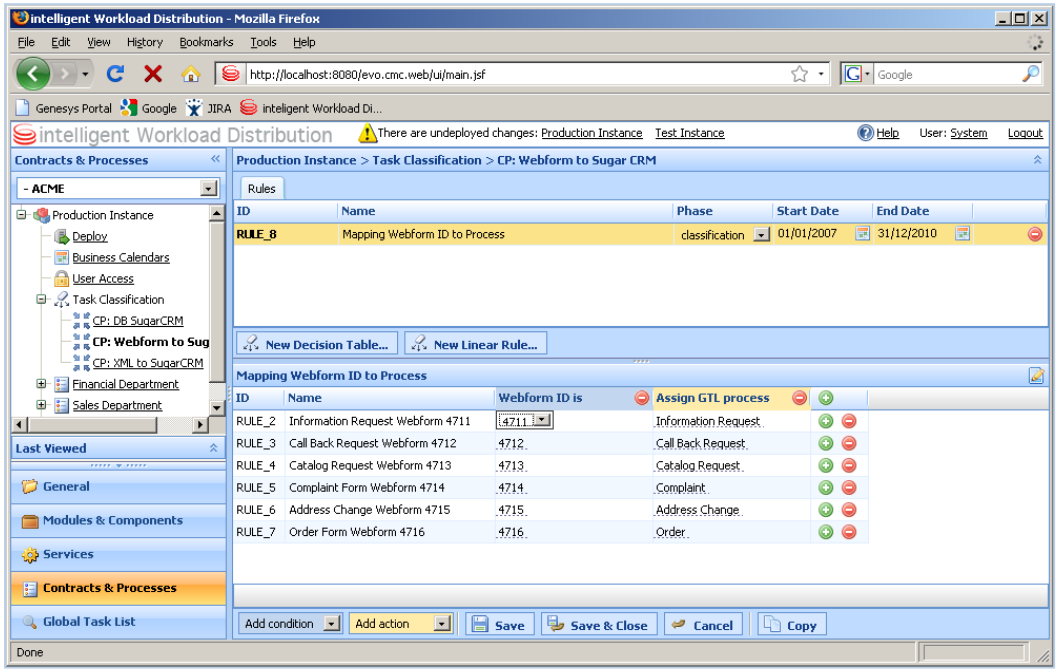

Figure 1: Centralized intelligent Workload Distribution configuration

Incremental Distribution

All intelligent Workload Distribution configuration changes are made offline and, upon completion, are distributed to iWD services in batches. This enables an automatic activation of all environment changes at once, avoiding any downtime that can be caused otherwise by a partially reconfigured system. It also enables a quick rollback to previous configuration versions, in case of any malfunction after distribution of new changes.

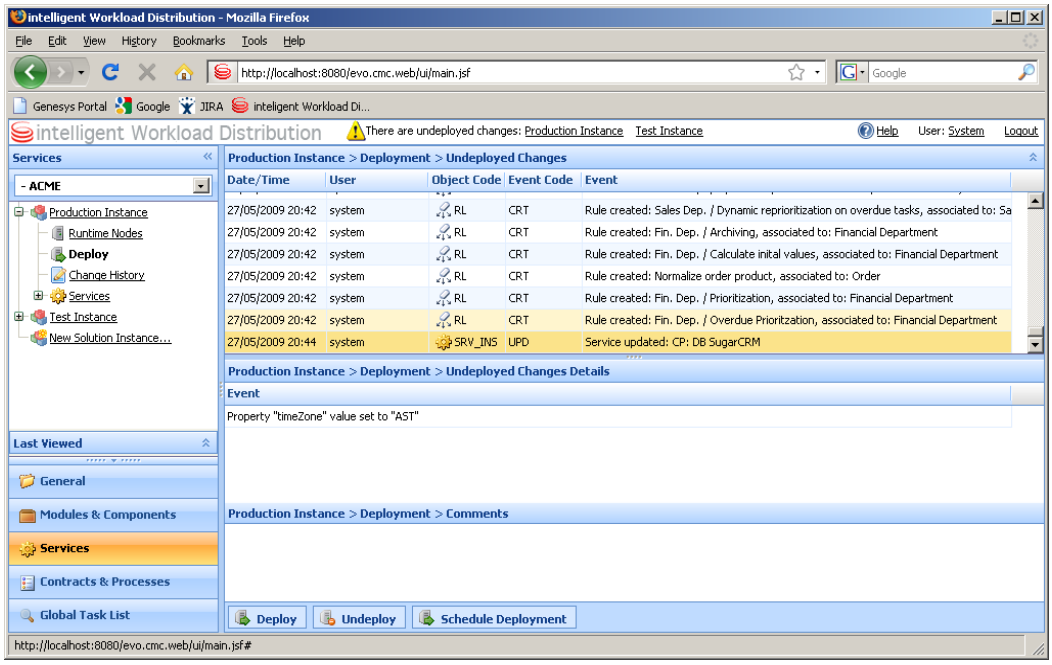

Figure 2: Incremental distribution of configuration

"Do not repeat yourself" principle

All configuration data objects, even if used by multiple iWD services, require single entry. Upon completion, information is distributed to all iWD services that require this configuration data. This process effectively eliminates all errors from duplicate data entry, while reducing time that is required implementing these changes.

Validation and Integrity Checks

All configuration data input into intelligent Workload Distribution is verified against several integrity rules. These rules control uniqueness, valid ranges of values, object-association principles, object-removal conditions, and other characteristics of configuration data that could otherwise cause the software to malfunction.

User Access Control

intelligent Workload Distribution offers comprehensive access control. Every iWD user has a password-protected account that is mapped to one or more security roles that are configured in iWD.

Permission to view or modify any system data is granted or denied, based on permissions assigned to the user's role(s). This ensures that only appropriate personnel have the access and authority to make changes to the system business or technical configuration, and to perform certain functions within the application. For example, a user may have permission to view tasks in iWD Manager but not permitted to hold or modify. Another user may have permission to modify business rules, but cannot delete.

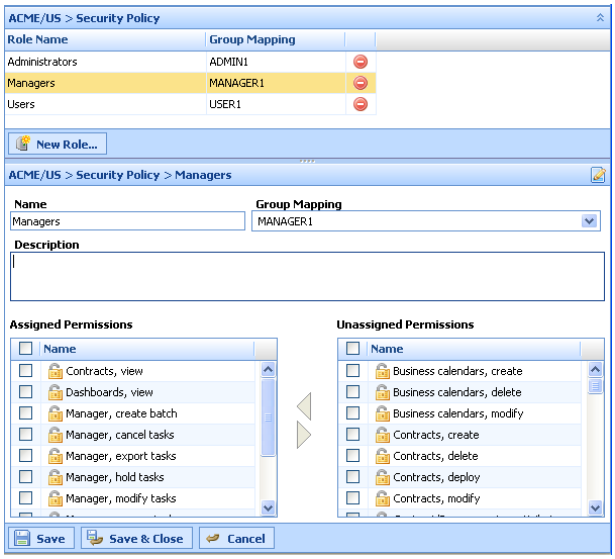

Figure 3: intelligent Workload Distribution security policy

System Administrators will typically be responsible for defining these permissions and roles in iWD and mapping to roles in the users' directory.

Change Control

Every configuration change that is performed in iWD is audited, which allows system administrators to see quickly what change was made and by whom. This level of change control applies to all objects in iWD, including business configuration objects like contracts, processes and rules, and technical configuration objects like services.

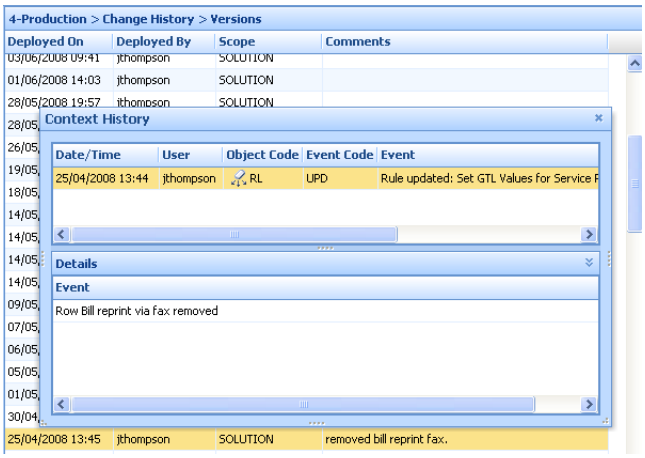

Figure 4: intelligent Workload Distribution configuration auditing and versioning

In addition, iWD automatically records each deployed change with the date/time and user who executed the deployment. Users can view changes to all objects, along with additional user notes on the reason/purpose for the deployment.

Management and Monitoring

System-management solutions can also contribute to improving system uptime by providing centralized and remote management, troubleshooting, and problem resolution. For all errors that are detected, but not automatically corrected, iWD provides integrated service-management capabilities to users, to quickly identify and correct these issues via the following:

- Centralized Management and Monitoring
- Logging services
- Notifications

Centralized Management and Monitoring

iWD incorporates a console for centralized service control, as well as monitoring for viewing the real-time status of service and for activating and deactivating services.

Logging Services

Each service in intelligent Workload Distribution implements a detailed logging. During a normal operation, log messages can set to a lower level of logging to ensure that log files are not growing unnecessarily in size. In case of a service issue or malfunction, activation of full logging for that service (or the entire system) provides system administrators with deep insight into the functioning of the system or service that will help in isolation of the root cause during troubleshooting. All log files can be accessed directly via the same centralized-management console, which allows monitoring of the overall service state.

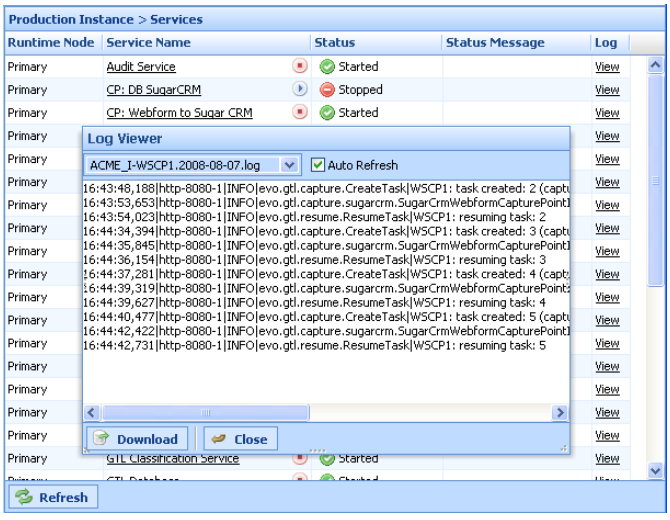

Figure 5: Service management and logging

Notifications

intelligent Workload Distribution supports the enabling of Simple Network Management Protocol (SNMP) traps for conditions that warrant administrative attention.

Support for integrated SNMP traps is scheduled for the upcoming iWD release. Until this release, current customers can leverage third-party adapters to set up traps, based on iWD logging messages. An example of such an adapter is the SNMPTrapAppender utility provided by Adventnet. Further information on this third-party component is available via the following link: http://www.adventnet.com/products/snmpadaptor/help/snmp_adaptor/sending_traps_for_log4j_log_messages.html#Overview.

Transactions

iWD is a fully transactional system that ensures 100 percent data consistency. Each operation in intelligent Workload Distribution is atomic, and is either successful completely or fully rolled back. The cause of the roll back can be via any one of the following types of failure:

- Software failure
- Hardware failure
- Communication failure

Communications

Queued Communication

intelligent Workload Distribution leverages message-queue-based communication protocols, wherever possible. This is advantageous for high-availability support, as it ensures delivery of data to the receiving party, even when it is down. The queue-based communication also ensures that the sender is not blocked by the unavailability of the receiver (unless a synchronous communication is required) and can process further requests, to maintain at least partial availability of the system.

Inside of iWD, all inter-communications among services (except for heartbeat monitoring) take place via an internal performance-optimized message queue. External systems can also be integrated via message-queue-based protocols (for example, via Websphere Message Queue services).

Connection Management

intelligent Workload Distribution implements active connection management for connections in which lower-level protocols are used, such as TCP. Examples of such

connections include databases and Genesys servers. Active connection management supports the following capabilities:

- Active disconnect detection, which includes polling of other party to ensure that the connection is alive
- Attempts at automatic reconnection in case of disconnect, until the connection is resumed
- Connection pooling.

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.

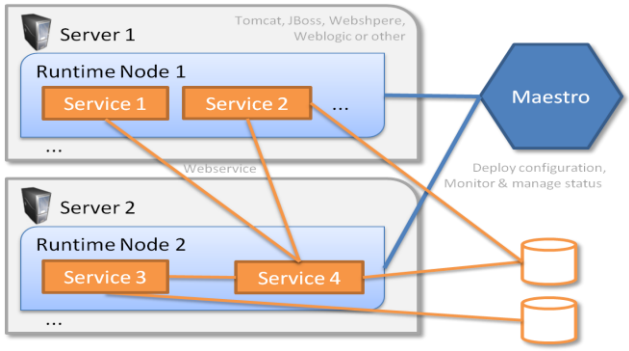

Figure 6: Distributed Deployment with centralized Management

In addition to this distributed architecture, a redundant solution typically comprises two services:

- A primary service that runs during normal operations
- A backup service that supports the primary service in case of failure, with a failover mechanism that ensures that the backup will take over from the primary service in case of service failure.

There are two different modes of redundancy support:

- Warm Standby
- Hot Standby

Warm Standby

The term warm standby is used to describe the redundancy mode in which a backup service is started in standby operation mode and ready to take over the operations of the primary service, when that service appears to be unavailable.

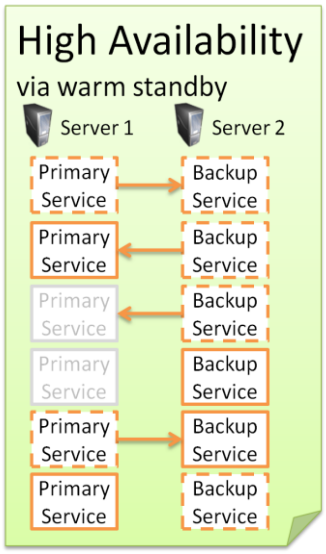

Figure 7: Warm Standby- Sequence between Primary & Back-Up Services

The backup service recognizes its role as a backup service, and will not become active until it detects that the primary service has become unavailable. In this case, it will switch to active operation mode and take over the processing from the primary service. When the primary service is active again, backup service returns to standby mode, and the primary service resumes processing.

The availability state of primary and backup are detected via bidirectional hear-beat monitoring over the standard HTTP protocol. There is no centralized external monitoring component in iWD, which minimizes possible points of failure.

Hot Standby

Hot standby is similar to warm-standby redundancy mode, but additionally it incorporates measures to avoid losing any data during a fail-over process by utilizing a buffer between the primary and backup services.

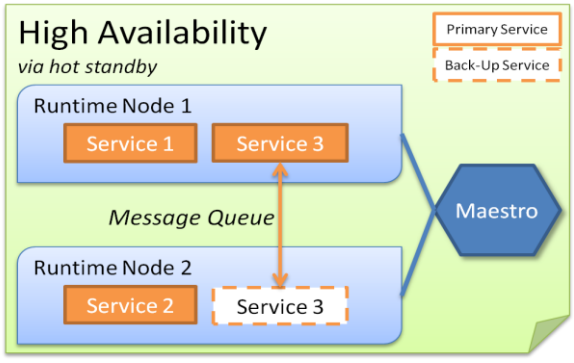

Figure 8: Hot Standby using Message Queue between iWD Services

This is when the backup service is working in a semi-active mode, maintaining the same communication connections as the primary for which it is a designated backup; however, the backup service does not act on the communication that is received over this connection – but only buffers. In the case of a failover, the backup service will first process data from the point at which the primary service left off.

Most of the intelligent Workload Distribution services rely on message-queue-based communications, and therefore, support hot-standby redundancy mode, without the need for a communications buffer; the queue is essentially providing the buffer service. One exception to this is Genesys Distribution Point, which dependents on a lower level TCP-based Open Media API communication protocol.

Redundancy Support Matrix

The following table lists intelligent Workload Distribution services and their supported redundancy mode or modes:

iWD Service	Redundancy Support
MQ Capture	Hot Standby
XML Capture	Hot Standby
Database Capture	Hot Standby
Classification Service	Hot Standby
Prioritization Service	Hot Standby
Genesys Distribution	Warm Standby Hot Standby
Statistics	Hot Standby
iWD Data Mart ETL	N/A