Genesys Info Mart 8.1

User's 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

Genesys and the Genesys logo are registered trademarks of Genesys Telecommunications Laboratories, Inc. All other company names and logos may be trademarks or registered trademarks of their respective holders.

The Crystal monospace font is used by permission of Software Renovation Corporation, www.SoftwareRenovation.com.

Technical Support from VARs

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

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 Program Guide*** 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: 81gim_us_07-2014_v8.1.401.00

Table of Contents

Preface

Welcome to the *Genesys Info Mart 8.1 User's Guide*. This document explains how to use data that is stored by Genesys Info Mart for contact center historical reporting. The document shows which dimension tables are associated with each fact table, describes validated interaction flows that are used by Genesys Info Mart, and explains how these interactions are represented in the Genesys Info Mart database tables.

In brief, you will find the following information in this guide:

- An overview of table data
- Descriptions of how data that is related to interaction-handling attempts, interaction resources, interactions, mediation segments, and contact attempts for Outbound Contact campaigns is populated
- Validated voice interaction flows
- Validated multimedia interaction flows
- Explanation of how to use the voice-of-data aspect of data lineage
- How dates and times of day are represented

The information is intended for end-users of Genesys Info Mart and is valid only for the Genesys 8.1 version of the software release.

Note: For versions of this document created for other releases of this product, visit the Genesys Documentation 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 193.

About Genesys Info Mart

Genesys Info Mart produces a data mart that you can use for contact center historical reporting.

Genesys Info Mart includes a server component, administration graphical user interface (GUI), and database. The Genesys Info Mart server runs a set of predefined jobs to:

- Extract data that has been gathered by Interaction Concentrator from data sources such as Configuration Server, T-Server, Interaction Server, and Outbound Contact Server. Genesys Info Mart stores this low-level interaction data, which is consolidated from Interaction Concentrator databases (Interaction Databases [IDBs]), in the Info Mart database.
- Transform the low-level interaction data and load it into a dimensional model (or star schemas) in the Info Mart database.

Genesys Info Mart can also be configured to host an aggregation engine that aggregates or re-aggregates the data, and populates Aggregate tables in the Info Mart database.

You query the fact and dimension tables in the dimensional model, using Structured Query Language (SQL), to obtain results that enable you to examine the data in detail, identify patterns, and predict trends for your organization.

Intended Audience

This guide is primarily intended for business users who want to query the data and for business-application developers who want to develop business-intelligence applications that query the data. The guide assumes that you have a solid understanding of database-management systems and structured query languages (such as SQL). Familiarity with CTI (computer-telephony integration) concepts, processes, terminology, and applications would also be helpful as would a basic understanding of the Genesys Framework—its architecture and functions.

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 Program Guide* for complete contact information and procedures.

Document Change History

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

New in Document Version v8.1.401.00

The document has been updated to support Genesys Info Mart release 8.1.4. The following topics have been added or changed since the previous release of this document:

Documentation Updates to Support New 8.1.4 Functionality

- The section on page 22 that describes special handling for "runaway strategies" (strategies which generate excessive numbers of unsuccessful attempts to route interactions to a handling resource) has been updated to include enhancements for multimedia transformation.
- The functionality provided by the `ignored-reason-codes` configuration option has been included in “The SM_RES_STATE_REASON_FACT Table” on page 43.
- Table 6 on page 55 has been updated to indicate that the name of the `PulledBackTimeout` role reason and technical result reason has changed to `PulledBack`, and and Table 9 on page 178 has been updated to indicate that there are new technical result combinations, `INCONFERENCE/REDIRECTED/PULLEDBACK` and `ROUTEDTO/REDIRECTED/PULLEDBACK`.

Documentation Enhancements and Corrections

- In the information about attached user data, a note on page 25 warns about potentially conflicting uses of Genesys-defined key-value pairs (KVPs).

New in Document Version v8.1.301.00

The document has been updated to support Genesys Info Mart release 8.1.3. The following topics have been added or significantly changed since the previous release of this document:

Documentation Updates to Support New Features and Software Modifications

- The description of high-cardinality user data in "User Data" on page 24 has been updated to include support, starting with release 8.1.201, for storing custom user-data facts as date/time data types.
- In Chapter 2, a new subsection, "Special Considerations for Long-Duration Sessions or States" on page 46, describes new and existing timeout functionality for agent login sessions and states.
- Table 6 on page 55 and Table 9 on page 178 have been updated to include the new technical result reasons, `ARCHIVED` and `CANCELED`, which are used in combination with the technical result of `COMPLETED` to identify when a handling resource places an interaction into an archive queue.

Documentation Enhancements and Corrections

- In Chapter 2, a new subsection, "Customer and Noncustomer Metrics" on page 26, clarifies the differences between the customer and noncustomer metrics in `INTERACTION_RESOURCE_FACT` (IRF) records.
- Misleading indications of customer time in Figure 34 on page 97 and Figure 36 on page 98 have been corrected.

New in Document Version v8.1.201.00

The document has been updated to support Genesys Info Mart release 8.1.2. The following topics have been added or significantly changed since the previous release of this document:

Documentation Updates to Support Software Modifications

- In the section about populating IRFs and dimensions, a new subsection on special handling for "runaway strategies" on page 22 replaces and updates information that was previously in a note.

Documentation Enhancements and Corrections

- Figure 1 on page 16 has been updated to include the `WORKBIN` dimension.
- In the section about populating IRFs and dimensions, a note on page 22 highlights an important difference between release 8.0 and release 8.1.
- A new subsection, "Limitation for Customer-Related Voice Activity" on page 29, describes a limitation for populating customer-related activity in the IRF table.
- In Table 4 on page 48, an incorrect statement about what happens to `Busy` state when Do-Not-Disturb (DND) is turned `on` has been corrected.
- In Chapter 4, Figure 61 on page 117 was moved out of the "IVR-Behind-Switch Call Flows" section into the "Universal Routing Assisted by IVR-Behind-Switch Call Flows" section.

- The multimedia interaction-flow diagrams in Chapter 5 on page 125 have been extensively revised, primarily to clarify the relationships between Interaction Queues, strategies, and virtual queues. In addition, new diagrams have been added to illustrate the results of multiple parallel or sequential virtual queues (see Figure 75 on page 137), of a blocking consultation (see Figure 77 on page 141), and of repeatedly unsuccessful routing attempts (see Figure 84 on page 150).

New in Document Version v8.1.101.00

The document has been updated to support Genesys Info Mart release 8.1.1. The following topics have been added or significantly changed since the previous release of this document:

- The bus matrix depicted in Figure 1 on page 16 has been updated to include the new ANCHOR_FLAGS dimension.
- In the section about "Populating Interaction Resource Data" on page 20:
  - A bullet item describes the new Interaction Resource Fact (IRF) fields (LAST_MEDIATION_SEGMENT_ID and RECEIVED_FROM_IXN_RESOURCE_ID) that link Mediation Segment Fact (MSF) and IRF activity and provide transfer details.
  - A bullet item describes the new functionality that supports agent and interaction thread metrics. The new functionality is further described in the bullet item about the ANCHOR_FLAGS dimension in the subsection, "Dimensions Associated with the IRF Table" on page 23.
- In the section about "Populating Interaction Facts and Dimensions" on page 31, a bullet item about MEDIA_SERVER_ROOT_IXN_ID and a new subsection, "Interaction Threads" on page 32, provide information about the new functionality that supports agent and interaction thread metrics.
- In the section about "Populating Mediation Segment Facts and Dimensions" on page 35:
  - Bullet items describe the new WORKBIN dimension and the new IXN_RESOURCE_ID and ENTRY_ORDINAL fields.
  - The description of the MEDIATION_DURATION field has been modified to reflect changed behavior.
- The description of how the SM_RES_SESSION_FACT table is populated (page 41) has been modified, to reflect discontinuation of the populate-sm-resource-session-facts configuration option in Genesys Info Mart 8.1.1.
- In the section about populating "The SM_RES_STATE_REASON_FACT Table" on page 43, an incorrect bullet item has been removed from the list of considerations that govern the choice of reason code that is reported in the table. The deleted item incorrectly stated that a reason code that starts later overrides an earlier reason code state.

- The new technical result (`AbnormalStop`) and associated technical result reasons (`AbnormalStopWhileRinging` and `AbnormalStopWhileQueued`) have been added to Table 6 on page 55 and to Table 9 on page 178. The description of the technical result reason `Cleared/Stopped` in Table 6 has been modified to exclude scenarios that are now reported as `AbnormalStop`.
- A new section, "Network Call Scenarios" on page 63, describes differences between Genesys Info Mart 7.6 and Genesys Info Mart 8.x in reporting resource roles and technical results in certain network call scenarios.
- In Chapter 4 on page 65, new call flow diagrams have been added to the inbound call flow examples. See Figures 18, 19, 20, and 21, starting on page 84.
- In Chapter 7, a note has been added to the end of the section about "Calendar Years and Week-Numbering Years" on page 167.
- Material has been modified throughout this guide to improve usability and accuracy and to reduce duplication in the Genesys Info Mart documentation suite. In particular:
  - The information about technical descriptors has been revised and extended. A new chapter, Chapter 3 on page 51, includes information that used to be in Chapter 2 on page 17, and a new Appendix on page 177 lists the technical descriptor combinations that Genesys Info Mart supports.
  - A new chapter, Chapter 8 on page 171, includes information about data-quality issues that used to be in Chapter 2 on page 17. Statements about data-source availability in this chapter (for example, in "Late Data" on page 173) have been changed to accommodate modifications in ETL processing.

Chapter

1 Genesys Info Mart Overview

Genesys Info Mart uses multidimensional modeling to create a constellation of star schemas. These star schemas create a database for storing contact center data that can be retrieved using queries. Star schemas support queries that speed the retrieval of the stored data. Querying the data helps you uncover trends, chart heavy usage times, and reveal patterns in your contact center. In this way, Genesys Info Mart can help you:

- Determine how to measure the efficiency of your contact center in comparison with targeted service goals.
- Determine how best to staff your contact center.
- Understand customer preferences and problem trends.

This chapter contains the following sections:

Genesys Info Mart Data

Genesys Info Mart 8.x extracts data from one or more Genesys Interaction Concentrator databases (Interaction Databases [IDBs]) and produces a data mart for contact center historical reporting. Genesys Info Mart yields data that is read-only and historical (representing some period of time).

The Info Mart database consists of the Global Interaction Database (GIDB) tables, fact and dimension tables (*dimensional model*), Merge Interaction

Database (MIDB) tables (used for voice interactions only), Control tables, the Staging area, and Temporary tables.

Note: GIDB provides the possibility for custom reporting or for drill-down reports from the dimensional model.

This guide focuses on the fact and dimension tables, as they are the primary sources of reporting data.

Fact Tables

Fact tables are the large tables in the middle of a star schema. They represent business measures—for example, how long customers waited in a queue, how long and how often agents put customers on hold, or how long agents talked to customers. Fact tables are surrounded by a set of slowly changing dimension tables. Fact tables represent a many-to-many relationship between dimensions; that is, there are many facts in a single fact table, and they are related to many dimensions in various dimension tables. Fact tables reference dimensions by using surrogate key columns.

Dimension Tables

Dimension tables describe the attributes of the associated fact table. For example, the dimensions that are related to interactions might include the date and time when each interaction started, the required skills for various service types requested by customers, and the value of various customers to the business.

Data Aggregation

An aggregation engine creates Aggregation tables and aggregates data in environments in which either Genesys Interactive Insights (GI2) reports or Reporting and Analytics Aggregates (RAA) package are deployed. These Aggregate tables are documented in the *Reporting and Analytics Aggregates Reference Manual*.

Subject Areas

Genesys Info Mart contains several subject areas that are of interest for contact center historical reporting—for example, the Interaction, Mediation Segment, or Resource Group subject area. Each subject area is a star schema. For more information about the subject areas and about the fact and dimension tables that are contained in each subject area, refer to the *Genesys Info Mart Reference Manual* for your RDBMS.

Bus Matrix

Figure 1 on page 16 maps the relationships between Genesys Info Mart fact tables and dimensions in a *bus matrix*.

The bus matrix represents dimensionality of fact tables in Genesys Info Mart as consolidated tabular views. It enables you to see the full dimensionality of each fact table easily.

Fact table names are listed in columns in the matrix; dimensions are listed in rows.

The matrix excludes the `TENANT` and `DATE_TIME` dimensions, which map to all fact tables. It excludes the media-specific interaction and interaction resource tables, as well as the `CTL_AUDIT_LOG` table (see page 19).

For information about all the fields that make up the facts and dimensions, refer to the *Genesys Info Mart8.1 Reference Manual* for your RDBMS. For information about aggregates, see the *Reporting and Analytics Aggregates Reference Manual*.

DIMENSIONS / FACT TABLES	CALLING_LIST_METRIC_FACT	CALLING_LIST_TO_CAMP_FACT	CAMPAIGN_GROUP_SESSION_FACT	CAMPAIGN_GROUP_STATE_FACT	GROUP_TO_CAMPAIGN_FACT	CONTACT_ATTEMPT_FACT	INTERACTION_FACT	INTERACTION_RESOURCE_FACT	IXN_RESOURCE_STATE_FACT	MEDIATION_SEGMENT_FACT	PLACE_GROUP_FACT	RESOURCE_GROUP_FACT	RESOURCE_SKILL_FACT	SM_RES_SESSION_FACT	SM_RES_STATE_FACT	SM_RES_STATE_REASON_FACT
ANCHOR_FLAGS								X								
ATTEMPT_DISPOSITION						X										
CALL_RESULT						X										
CALLING_LIST	X	X				X										
CAMPAIGN	X	X	X	X	X	X										
CAMPAIGN_GROUP_STATE				X												
CONTACT_INFO TYPE						X										
DIALING MODE						X										
GROUP_			X	X	X	X					X	X				
INTERACTION_DESCRIPTOR								X								
INTERACTION_RESOURCE_STATE									X							
INTERACTION_TYPE							X	X	X	X						
MEDIA_TYPE						X	X	X	X	X				X	X	X
PLACE						X		X	X		X					
RECORD_STATUS						X										
RECORD_TYPE						X										
RECORD_FIELD_GROUP1						X										
RECORD_FIELD_GROUP2						X										
REQUESTED_SKILL								X								
REQUESTED_SKILL_COMBINATION								X								
RESOURCE_						X		X	X	X		X	X	X	X	X
RESOURCE_GROUP_COMBINATIO								X		X				X	X	X
RESOURCE_STATE								X							X	X
RESOURCE_STATE_REASON																X
ROUTING_TARGET								X								
SKILL													X			
STRATEGY								X								
TECHNICAL_DESCRIPTOR								X		X						
TIME ZONE						X										
WORKBIN										X						

Figure 1: Bus Matrix of Fact and Dimension Tables

Chapter

2 Populating Genesys Info Mart Data

This chapter describes how Genesys Info Mart populates the data in the Genesys Info Mart database. You need this information in order to create meaningful queries for business purposes, as well as to interpret query results correctly.

This chapter contains the following sections:

For information about specific columns in Genesys Info Mart tables, see the *Genesys Info Mart 8.1 Reference Manual* for your particular relational database-management system (RDBMS).

Bringing Data into Info Mart

Extract, transform, and load (ETL) is performed by two main jobs, `Job_ExtractICON` and `Job_TransformGIM`.

Deployments in which Genesys Interactive Insights (GI2) or Reporting and Analytics Aggregates (RAA) is installed also use `Job_AggregateGIM`.

`Job_ExtractICON` extracts new and changed data from Interaction Databases (IDBs) and stores it in the GIDB tables, as discussed in "Populating Low-Level Details" on page 18.

`Job_TransformGIM` transforms the data from GIDB into the dimensional-model (fact and dimension) tables.

`Job_AggregateGIM` calculates or recalculates metrics and stores them in the aggregate tables in the Info Mart database, based on the data that was added or changed during the last transformation run.

Note: For more information about the Genesys Info Mart jobs, see the *Genesys Info Mart 8.1 Operations Guide*. For detailed information about Genesys Info Mart functioning, see the *Genesys Info Mart 8.1 Deployment Guide*. For detailed information about the aggregation process, see *Reporting and Analytics Aggregation documentation set*.

Populating Low-Level Details

The Global Interaction Database (GIDB) is an area within the Genesys Info Mart database schema in which the low-level interaction data from any number of IDBs is consolidated for further processing.

To populate GIDB, Genesys Info Mart Server extracts data from one or more source IDBs. For voice-interaction records, the merge operation links all records that are related to the same interaction, in both single-site and multi-site deployments. The server loads all extracted (and, if applicable, merged) data into GIDB.

The GIDB tables:

- Represent a subset of IDB tables, to better align the lowest level of data details in Genesys Info Mart with the Interaction Concentrator model.
- Provide low-level details about a call, party, and party history for voice and multimedia interactions in the `GIDB_G_CALL`, `GIDB_G_PARTY`, and `GIDB_G_PARTY_HISTORY` tables, respectively.
- Store all extracted records that are necessary for Genesys Info Mart reporting purposes from various IDBs, to gather coherent reporting data at the lowest level of detail from the entire contact center in a single data warehouse.
- Use special fields to indicate from which IDB data was extracted.
- Store the data for as long as it is required by customers after Genesys Info Mart further processes (transforms) GIDB data.

Note: The term *voice interactions* refers to traditional telephony calls. The term *multimedia interactions* refers to interactions that are processed through the Genesys eServices/Multimedia solution, including 3rd Party Media interactions.

For information about the meaning of other terms, such as *data sources,* see the section about terminology conventions in the Overview chapter of the *Genesys Info Mart 8.1 Deployment Guide*.

Genesys Info Mart Server uses the low-level details data from GIDB tables to produce data that is suitable for end-user reports and to populate the fact and dimension tables that compose the Info Mart dimensional model.

The *Genesys Info Mart 8.1 Reference Manual* for each supported RDBMS provides a list of GIDB tables. The meaning of the data in each row within a given GIDB table is the same as in the corresponding IDB record. For example, `GIDB_GC_PLACE` table in the Info Mart database corresponds to the `GC_PLACE` table in IDB. Refer to the *Interaction Concentrator 8.x Physical Data Model* document for your RDBMS for information about the data stored in corresponding GIDB tables.

The DATE_TIME Dimension

The `DATE_TIME` dimension enables facts to be described by attributes of calendar date and 15-minute time interval.

All interaction-related fact tables use only the `DATE_TIME` time dimension. No other time-dimension fields are used.

Note: Only UTC timestamps are used in the interaction-related fact tables.

By default, a single `DATE_TIME` table is configured, but you can set up multiple calendar tables. For example, you might need to support multiple time zones. For details on how to configure multiple `DATE_TIME` tables, see the *Genesys Info Mart 8.1 Deployment Guide*.

The `DATE_TIME` dimension is discussed in greater detail in Chapter 7, "Representing Dates and Times of Day," on page 165.

The CTL_AUDIT_LOG Dimension Table

The `CTL_AUDIT_LOG` dimension table contains data for all transactions that are committed by Genesys Info Mart. This table replaces the Genesys Info Mart 7.x `AUDIT_` dimension table. Instead of service fields such as `ROW_CREATED`

and ROW_UPDATED appearing in all tables, the CTL_AUDIT_LOG table contains audit information for all records.

All fact table records now contain pointers (CREATE_AUDIT_KEY and UPDATE_AUDIT_KEY) to the relevant CTL_AUDIT_LOG table row.

Each row represents a logical transaction that is committed by Genesys Info Mart, identifying the ETL job involved in the transaction and including the minimum and maximum DATE_TIME values (which give the date-time range for the data that is committed in the transaction), and providing the processing status (an internal indicator of the kind of data that is processed).

Populating Interaction Resource Data

Genesys Info Mart stores interaction resource facts in the INTERACTION_RESOURCE_FACT (IRF) table, one of the core tables that is supplied in Genesys Info Mart.

This table facilitates the creation of reports and serves as one of the primary tables from which aggregation tables are populated. (See the *Reporting and Analytics Aggregation documentation set* for details on aggregation tables.)

Genesys Info Mart creates IRFs to represent the involvement of a contact center resource of interest in an interaction. *Resources of interest* in the IRF context are:

- Handling resources—agents, self-service IVRs, DNs without an agent, and multimedia strategies that handle an interaction (for example, a strategy that sends an AutoResponse). Genesys Info Mart creates a row in the IRF table whenever a new interaction or a new attempt to handle an existing interaction has been started, or when an interaction arrives at a handling resource.
- Mediation resources—such as queues, routing points, and nonself-service IVRs— in which the interaction ends.

The IRF table supplies a single row within the Genesys Info Mart schema, which simplifies the SQL needed to generate reports on the resources that handle interactions within the contact center.

Each IRF represents:

- The contiguous time span of the association between the resource and the interaction
- The particular role played by the resource (the *resource role*)
- The result of the association from the perspective of the resource (the *technical result*)

IRFs are created for completed voice interactions and for both completed and active multimedia interactions.

The IRF table:

- Simplifies report queries by integrating conference and consultation durations into the original handling-resource row.
- Summarizes the total queue, routing point, and IVR wait times prior to the handling resource and stores the summary data with the handling-resource row in separate columns.
- Stores response duration per routing attempt, in addition to the initial routing sequence.
- Records the state of the resource immediately prior to involvement in the interaction, thus enabling reporting of interactions received or initiated during an `AfterCaLLwork` or `NotReady` agent state.
- Links the IRF to associated MSFs, to provide information about:
  - The last mediation segment that was involved in the interaction, regardless of whether the interaction was distributed to a handling resource (`LAST_MEDIATION_SEGMENT_ID`)
  - The interaction resource that originated a transfer, conference, or voice consultation (`RECEIVED_FROM_IXN_RESOURCE_ID`)

  Together with fields in the MSF table that link associated MSFs to the IRF, these fields enable downstream reporting applications to report on transfer details and queue activity, including interactions that were abandoned or cleared in virtual queues.
- Provides thread metrics and agent-unique metrics for an interaction or thread, which identify whether the IRF represents:
  - This agent's first participation in the interaction (where "this agent" refers to the agent resource that is the subject of the IRF record)
  - This agent's first participation in a reply within the interaction
  - This agent's first participation in the interaction thread
  - This agent's first participation in a reply within this interaction thread
  - The first participation by any handling resource in the interaction thread
- For voice calls, provided that T-Server and the Interaction Concentrator server (ICON) have been configured to provide the required data, indicates whether a given resource initiated release of the call.

Genesys Info Mart uses the following additional tables to support the IRF table:

- The `IXN_RESOURCE_STATE_FACT` table contains all the individual states, durations, and interval clips for each state the interaction-fact resource was in during the interaction.

- The INTERACTION_RESOURCE_STATE dimension table contains the states defined for the resource that is handling the interaction.

Note: For detailed information about the columns in the IRF table, see the *Genesys Info Mart Reference Manual* for your RDBMS.

Populating Interaction Resource Facts and Dimensions

The following sections describe how Genesys Info Mart populates IRFs.

Each IRF row includes all prior queue, routing point, and IVR (nonself-service) counts and durations that were part of the distribution of the interaction to the resource.

Note: Population of the MEDIATION_SEGMENT_ID and MEDIATION_RESOURCE_KEY fields, which identify the mediation resource that distributed the interaction to the IRF resource, changed between release 8.0 and release 8.1. Starting with release 8.1, information is propagated to these fields from other IRFs in certain scenarios. For more information, see the field descriptions in the *Genesys Info Mart 8.1 Reference Manual*.

IRFs represent either the processing of interactions by handling resources (such as agents, self-service IVRs, and extensions/positions without associated agents) or unsuccessful attempts to reach such a handling resource (resulting in the interaction being abandoned in queue or abandoned in routing).

The grain of the fact is an accumulating snapshot of the contiguous participation of a contact center handling resource in interaction processing, including time spent wrapping up the interaction. Movement of a resource from one call to another does not cause creation of a new IRF, but is accumulated in a single fact. For example, when the transferredTo resource in a transfer scenario is moved from a consult call to the original call, this movement is represented in a single fact.

However, if a handling resource is participating in parallel calls, the resource is represented by two separate facts. For example, in a consultation call scenario there are two facts for the consulting resource, one for the existing call and one for the consultation call.

Special Handling for “Runaway Strategies”

Special logic protects Genesys Info Mart from being overwhelmed by strategies that cause very large quantities of Party, Virtual Queue, and Party History records in IDB. In most cases, having very large numbers of parties and virtual queues involved in a single interaction results from inappropriate strategies, which generate excessive numbers of unsuccessful attempts to route

interactions to a handling resource. For example, a strategy might be configured to pull a batch of multimedia interactions from an Interaction Queue, attempt to route the interactions, place the interactions that it was not able to route back into the Interaction Queue, and retry at 1-second intervals.

In these "runaway strategy" scenarios, the transformation job abbreviates the representation of unsuccessful routing attempts.

Genesys Info Mart's handling of "runaway strategy" scenarios changed between the initial 8.1.0 release and release 8.1.103.03, and again between 8.1.3 releases and release 8.1.4:

- To prevent performance and out-of-memory issues with "runaway strategies" in releases earlier than 8.1.4, Genesys Info Mart used to artificially limit how much of the data associated with a single interaction would be selected for transformation in a single chunk. In releases 8.1.103.03 through 8.1.3.x, the `max-parties-per-call` configuration option enabled users to specify the limits that trigger special handling. As a result of a more incremental approach to multimedia transformation in release 8.1.4, it is no longer necessary to artificially limit transformation chunk size; therefore, the `max-parties-per-call` option was discontinued in release 8.1.4. For more information, see the description of the `max-parties-per-call` configuration option in the *Genesys Info Mart 8.1 Deployment Guide*.
- The incremental transformation enhancements also result in improved data quality as a result of changes in the way that queue and routing point metrics for "runaway strategy" interactions are populated in IRF records. For more information, see the *Genesys Info Mart 8.1 Reference Manual*.

Genesys recommends that users in large-scale, production-level environments evaluate their strategy configurations, to minimize the risks to data quality by ensuring that their environments are not susceptible to these types of scenarios.

Dimensions Associated with the IRF Table

- Interaction-resource fact start and end dates and times are stored as UTC timestamps (`START_TS` and `END_TS`) and as references to the `DATE_TIME` dimension (`START_DATE_TIME_KEY` and `END_DATE_TIME_KEY`).

  Media-neutral counts and durations are provided. These categorize the time spent on various activities, such as time spent in a queue, time spent handling the interaction, and time spent wrapping up the interaction. Because not all IRFs involve a customer directly, separate counts and durations are included to reflect the time that the customer spent waiting versus being helped.

> **Note:** For more information about how Genesys Info Mart represents dates and times of day, see Chapter 7 on page 165.

- The RESOURCE_ dimension indicates the routing point, queue, IVR, or agent that either initiated or handled this resource fact.

> **Note:** The RESOURCE_ dimension actually has two references, RESOURCE_KEY and MEDIA_RESOURCE_KEY, which typically refer to the same resource. The following are exceptions:
> - For IVRs, RESOURCE_KEY is for the IVR Application Name and MEDIA_RESOURCE_KEY for the associated DN.
> - For Agents, RESOURCE_KEY is for the Agent, and MEDIA_RESOURCE_KEY for the associated DN.

- The PLACE dimension indicates the place at which the IRF was processed.
- The TENANT dimension identifies the tenant of the resource.
- The TECHNICAL_DESCRIPTOR dimension identifies the resource role and technical result of the IRF. For information about the resource roles and technical results for interaction resources, see Chapter 3 on page 51.
- The INTERACTION_DESCRIPTOR dimension identifies the customer segment (indicating the value of the customer), the type of service being requested, and the business result of the IRF.
- The STRATEGY dimension identifies the Genesys routing strategy or IVR application that processed the IRF.
- The ROUTING_TARGET, REQUESTED_SKILL, and REQUESTED_SKILL_COMBINATION dimensions indicate the Genesys Universal Routing Server's activities by identifying the target that was selected and the list of skills that were required to process the IRF.
- The CUSTOMER dimension represents the ID of the customer that is involved in the interaction.
- The ANCHOR_FLAGS dimension supports metrics about unique agent participation in an interaction or thread. For related information, see "Interaction Threads" on page 32.

User Data

As previously indicated, many interaction attributes are formally modeled. However, deployment-specific attributes, in the form of *user-defined attached data*, are also represented in the model.

Genesys Info Mart provides unified user-data processing from both call-related EventUserEvents and call-based TEvents, with flexible data storage that you can configure according to the number and types of user data captured in your contact center environment. A customizable database schema enables you to treat each key-value pair (KVP) field as a fact, a dimension, or both, and to store user data KVPs in a configurable number of user data dimensions and facts that are associated with core fact tables. Genesys Info Mart also processes

the user data that arrives after call completion and updates call records accordingly.

User data can be stored as facts or dimensions. There are two kinds of user data:

- *High-cardinality user data*—Data for which there can be a very large number of possible values. A Customer ID number is an example of high-cardinality user data. You can configure Genesys Info Mart to store the KVP value as a character data type, as a numeric data type, or, starting with release 8.1.201, as a date/time data type.
- *Low-cardinality user data*—Data for which there is a limited range of possible values. Customer Segment, Service Type, and Service Subtype are good examples of low-cardinality user data. For example, in a `CUSTOMER` table with a column named `NEW_CUSTOMER`, this column would contain only two distinct values, `Y` or `N`, which respectively denote whether the customer was new or not. Because only two possible values are held in this column, its cardinality type is low cardinality.

High-cardinality user data is stored as facts. Low-cardinality user data is most efficiently stored as dimensions. You can create up to 800 custom low-cardinality user data dimensions. The only limits on the quantity of high-cardinality user data that you store are performance based. There are no absolute limits.

High-cardinality user data requires only a single join from the IRF table. Low-cardinality user data that is stored as dimensions require two joins, one to the `CTL_UDE_KEYS_TO_DIM_MAPPING` table and another to the dimension table.

You can use the same KVP as both fact and dimension. Genesys provides templates for you to configure your own User Data keys.

Note: Be aware that other Genesys applications might use Genesys-defined KVPs. Depending on your reporting needs, you might need to co-ordinate use of Genesys-defined KVPs in your deployment by modifying the behavior of your routing strategies or, on the Genesys Info Mart side, changing the propagation rule for a particular KVP or creating a custom KVP.

For example, by default, the Genesys gateway application for Web Real-Time Communication (WebRTC) uses the `ServiceType` KVP with a value of `WebRTC` to indicate that the interaction is a WebRTC call. By default, Genesys Info Mart will store `INTERACTION_DESCRIPTOR.SERVICE_TYPE=WebRTC`, unless a routing strategy subsequently attached the `ServiceType` KVP with a different value during the call.

Customer and Noncustomer Metrics

Each IRF record includes numerous `*_COUNT` and `*_DURATION` metrics. There are two categories of metrics:

- Customer metrics (prefixed by `CUSTOMER_`), which reflect the "customer experience"—that is, how the customer was treated during an interaction
- Noncustomer metrics (not prefixed by `CUSTOMER_`), which reflect the "handling resource experience"—that is, how the contact center's handling resources spent time in an interaction

For detailed descriptions of the metrics, see the *Genesys Info Mart 8.1 Reference Manual* for your RDBMS.

Table 1 on page 26 summarizes which party is considered to be the customer and which is the handling resource for various types of interactions.

Table 1: "Customer" and Handling Resource, by Type of Interaction

Type of Interaction	"Customer"	Handling Resource
Inbound	The external party that initiated the interaction to the contact center	The contact center party that receives the interaction
Internal	The internal party that initiated the interaction within the contact center	The contact center party that receives the interaction
Outbound	The external party that is contacted by the contact center	The contact center party that initiated the interaction or, in the case of Outbound Contact, that was offered the interaction for handling
Consultation[a]	None	Both contact center parties that are involved in the consultation: • The party that initiates the consultation • The party that receives the consultation

a. Strictly speaking, a consultation (called *collaboration* for e-mail) is not itself a type of interaction; it occurs within the context of an interaction.

Customer Metrics

IRFs are created only for contact center resources. As shown in Table 1, the customer is generally an external party, for whom no IRF is created. Customer metrics accrue on the IRF for the handling resource, to show the customer experience alongside the noncustomer (handling resource) experience in the same IRF.

In this document, the time that the customer is considered to be present in the interaction is referred to as *customer time*.

Voice

For voice calls, customer time accrues on the IRF for the party that is considered to be the handling resource (as indicated in Table 1), as long as the party that is considered to be the customer is present in the context of the IRF. (For the IRF that represents an outbound or initiated Outbound Contact call, customer dial time accrues even though the customer is not yet present on the call.) Customer time stops accruing at the moment the party that is considered to be the customer releases or is released from the call.

For consultations, as shown in Table 1, both the initiator and the receiver of the consultation are handling resources. There is no customer present on the consultation, and no customer time accrues on the IRFs related to the consultation.

For graphical representations of customer time in various call topologies, see the call-flow diagrams in Chapter 4 on page 65.

Multimedia

For multimedia interactions, as for voice calls, customer time accrues on the IRF for the party that is considered to be the handling resource (as indicated in Table 1 on page 26), However, for multimedia interactions, unlike for voice calls, the notion of the customer being present often does not apply. In order for Genesys Info Mart to represent the customer experience for multimedia interactions, all time that the handling resource spends handling a multimedia interaction is considered to be customer time, except for initiated and received consultations (or e-mail collaborations).

For graphical representations of customer time in various interaction topologies, see the interaction-flow diagrams in Chapter 5 on page 125.

Noncustomer Metrics

Noncustomer metrics accrue on all types of IRFs. Noncustomer metrics are divided into separate “buckets” that represent the possible different phases of an interaction—interaction initiated or offered, initiated consult, received consult, post-consult transfer, conference initiated, conference joined.

Reporting Implications

Customer and Noncustomer Metrics Not Necessarily Equal

The counterpart customer and noncustomer metrics (for example, `CUSTOMER_TALK_DURATION` and `TALK_DURATION`) accrue in parallel within a given

IRF, but you cannot assume that they will be equal. The respective values of the metrics depend on:

- The behavior of the parties in the specific interaction. For example, in a voice call topology that includes a conference between the customer and two agents, the customer and noncustomer representations of talk duration will not be the same if the handling resource continues on the conference after the customer hangs up.
- The type of interaction. For example, in the IRFs for the initiator and the receiver of a consultation, `CUSTOMER_TALK_DURATION` will be `0` (zero), while `CONS_INIT_TALK_DURATION` and `CONS_RCV_TALK_DURATION`, respectively, will be nonzero.

Overlapping IRFs

For noncustomer metrics, the "buckets" that represent the separate phases of an interaction do not overlap within a single IRF, but they can overlap for parallel IRFs. For example, when the handling resource on an inbound interaction initiates a consultation, there will be two parallel IRFs with overlapping `HOLD_DURATION` and `CONS_INIT_TALK_DURATION` metrics.

Recommendations and Tips

Observe the following guidelines:

- Do not try to combine customer and noncustomer metrics.
- Do not expect that the sum of noncustomer metrics will equal the sum of customer metrics.
- To avoid double-counting time from overlapping IRFs for a given agent, do not combine IRFs that represent initiated consultations with other IRFs for the same agent. Instead, use initiated-consultation IRFs to calculate consultation metrics separately.
- For voice calls, use customer metrics when the reporting focus is on the customer experience. Use noncustomer metrics when the focus is on the activities of the handling resource(s), regardless of whether the customer party is present on the call.
- For multimedia interactions, use customer metrics when you do not want to consider collaboration (consultation) time. Use noncustomer metrics when you want to include collaboration time.

Useful Filters

The `HANDLE_COUNT` and `CUSTOMER_HANDLE_COUNT` fields are useful to identify various interaction scenarios. For example, Table 2 summarizes the different

results in these fields for the resource that is the subject of the IRF (*IRF resource*), for a number of common scenarios.

Table 2: HANDLE_COUNT and CUSTOMER_HANDLE_COUNT Values

Scenario	Type of Interaction	HANDLE_ COUNT	CUSTOMER_ HANDLE_ COUNT
Any outbound interaction initiated by an IRF resource	Outbound	1	1
Any interaction offered to an IRF resource that is accepted (answered) by that IRF resource	Inbound, internal, or outbound	1	1
Any interaction offered to an IRF resource that is not accepted by that IRF resource (for example, redirected or abandoned)	Inbound, internal, or outbound	0	0
Any consultation initiated by an IRF resource	Consultation[a]	1	0
Any consultation offered to an IRF resource that is accepted (answered) by that IRF resource	Consultation[a]	1	0
Any consultation offered to an IRF resource that is not accepted by that IRF resource (for example, redirected or abandoned)	Consultation[a]	0	0

a. Strictly speaking, a consultation (collaboration) is not itself a type of interaction; it occurs within the context of an interaction.

Limitation for Customer-Related Voice Activity

In general, the IRF table is populated in a way that enables downstream reporting applications to distinguish customer-related activity, including transfers and conferences, from internal agent–related activity—for example, in data and metrics such as the Technical Descriptor, `CONFERENCE_INITIATED_COUNT`, and `CUSTOMER_HANDLE_COUNT`.

However, the population of the dimensional model breaks down for consult/transfers and consult/conferences that occur within an existing conference: When a voice interaction contains a conference that involves a customer and more than one agent, and one of those agents initiates a subsequent transfer or conference out of the first conference, there is no clear way to reliably determine whether the customer is still present when the

subsequent transfer or conference occurs. Therefore, metrics such as count of customer-related transfers or count of customer-related conferences cannot be calculated reliably, although the equivalent metrics for internal agent–related transfers or conferences can.

Abandoned and Terminated Interactions

To represent every interaction in the IRF table, rows are created to represent attempts to reach a resource of interest. These rows contain data about queues, routing points, and routing queues in which the interaction has been abandoned in the distribution device by the customer, during a consultation, or during an internal call that was initiated by a resource of interest.

Abandoned Interactions

Abandoned interactions are identified as interactions in which the last resource that was involved was not a handling resource.

In such cases a row is created to represent an attempt to reach another handling resource. This IRF row contains data from all prior related mediation device segments that were involved with the attempt to reach another handling resource.

Interactions Terminated in a Mediation IVR or DN (No IVR or Agent Resource Association)

A *mediation IVR* in the context of the IRF table is an IVR resource that is not considered to be self-service because the IVR application (or a URS strategy on its behalf) did not set attached data to indicate self-service. An interaction that terminates in a mediation IVR is considered to be abandoned.

Populating Interaction Data

Genesys Info Mart creates interaction facts (IFs) to link together all facts related to a given interaction. IFs represent interactions from the perspective of the customer experience. For example, Genesys Info Mart represents every new inbound or outbound interaction as a new IF row; however, for multimedia interactions, an inbound interaction and an associated outbound reply are represented in the same IF.

Each interaction fact represents:

- The time span of the overall interaction
- Information that identifies the interaction parties
- Service indicators

Interaction facts can also be linked to the user data extension tables through keys.

Interaction Fact Table

Genesys Info Mart stores both voice and non-voice interaction facts in the INTERACTION_FACT table.

For detailed information about the columns in the Interaction Fact table, see the *Genesys Info Mart Reference Manual* for your RDBMS.

Populating Interaction Facts and Dimensions

Genesys Info Mart populates voice and multimedia interactions in the following ways:

- The TENANT dimension is inherited from the underlying IRF that has the lowest ordinal. This is the first resource facts that was created for the interaction, and it generally has the earliest start time. In a network routing solution, all underlying network and premise facts are considered. If premise facts exist, the TENANT dimension is the tenant of the first premise fact; otherwise, the TENANT dimension is the tenant of the first network fact.
- The INTERACTION_TYPE and MEDIA_TYPE dimensions are inherited from the underlying IRF that has the lowest ordinal. This is the first resource fact that was created for the interaction, and it generally has the earliest start time. In a network routing solution, all underlying network and premise facts are considered.

  Note: Any multimedia interaction subtype that you have configured in your environment but that is new to Genesys Info Mart is automatically added to the INTERACTION_TYPE table. Once it has been added, you can choose to have Genesys Info Mart disregard that subtype for all future transformation jobs by setting the appropriate value for the IGNORE field. By default, Genesys Info Mart transforms all interactions that have the newly added subtype.

  New media types are also automatically added as Genesys Info Mart encounters them. By default, interactions that are associated with new media types are transformed as offline interactions. To set them as online interactions, enter the appropriate value in the IS_ONLINE field in the MEDIA_TYPE table.

  For details, see the *Genesys Info Mart 8.1 Deployment Guide*.

- The MEDIA_SERVER_ROOT_IXN_ID acts as a thread ID for interactions that are a continuation of a thread. For more information, see "Interaction Threads".

Interaction Threads

Each customer interaction is represented in Genesys Info Mart with a new IF, but it is possible that different customer interactions are associated with one another. For example, a new inbound interaction from a customer might be a reply to a previous agent reply to another inbound interaction from that customer. A collection of related interactions is referred to as a *thread.* Genesys Info Mart indicates this thread relationship by showing the root interaction in the IF record for a descendant interaction.

- The `MEDIA_SERVER_ROOT_IXN_ID` identifies the IF that is considered to be the root (original) interaction in the thread. Population of this field depends on Genesys Info Mart's tracking of the thread, which is affected by the configured thread inactivity timeout (the `max-thread-duration-after-inactive-in-days` option). For an example, see Table 3.
- `MEDIA_SERVER_ROOT_IXN_GUID` identifies the root interaction GUID, as reported by ICON as the `ROOTIRID`.

Table 3 provides a sample e-mail scenario to indicate how Genesys Info Mart tracks the thread relationships for three related IFs, which have the same root interaction GUID: Two of the related IFs are considered to belong to the same thread, but the third one, which occurs after the thread inactivity interval has expired, is considered to start a new thread.

Table 3: Sample Thread Scenario

Date	E-Mail Interaction	Interaction and Thread IDs in IF
June 1	InboundNew: *Please send me 100 of ItemX and 50 of ItemY. My account number is 1234.*	`INTERACTION_ID`: 161 `MEDIA_SERVER_ROOT_IXN_ID`: Null `MEDIA_SERVER_ROOT_IXN_GUID`: Null `MEDIA_SERVER_IXN_GUID`: `ROOTIRID-1`
June 2	OutboundReply 1a from Agent1:[a] *Your account has been billed. Your order number is ZZ001.*	
June 2	OutboundReply 1b from Agent1: *Your order ZZ001 has been shipped.*	
June 2	InboundCustomerReply (to OutboundReply 1a): *There is a mistake in the billing details for order ZZ001.*	`INTERACTION_ID`: 174 `MEDIA_SERVER_ROOT_IXN_ID`: 161 `MEDIA_SERVER_ROOT_IXN_GUID`: `ROOTIRID-1`
June 15	OutboundReply 2a from Agent2:[a] *Correction has been made. Here are the correct billing details...*	

Table 3: Sample Thread Scenario (Continued)

Date	E-Mail Interaction	Interaction and Thread IDs in IF
Assuming no further activity on this thread and `max-thread-duration-after-inactive-in-days=30`, Genesys Info Mart closes the thread on July 16.		
July 20	InboundCustomerReply (to OutboundReply 1b): *Order ZZ001 arrived. We need 50 more of ItemY. Same account.*	`INTERACTION_ID`: 249 `MEDIA_SERVER_ROOT_IXN_ID`: Null `MEDIA_SERVER_ROOT_IXN_GUID`: Null **Note:** The `MEDIA_SERVER_ROOT_IXN_ID` and `MEDIA_SERVER_ROOT_IXN_GUID` are null because Genesys Info Mart considers this to be a new thread, after the previous thread (associated with `INTERACTION_ID=161`) was closed because of the inactivity timeout.
July 21	OutboundReply 3a from Agent1:[a] *Your account has been billed. Your order number is ZZ002.*	
July 22	OutboundReply 3b from Agent1: *Your order ZZ002 has been shipped.*	
July 22	InboundCustomerReply (to OutboundReply 2a): *Based on the price of ItemY in order ZZ002, there is still a problem with the billing for order ZZ001.*	`INTERACTION_ID`: 265 `MEDIA_SERVER_ROOT_IXN_ID`: 249 `MEDIA_SERVER_ROOT_IXN_GUID`: ROOTIRID-1
July 22	OutboundReply 3c from Agent1:[a] *You are correct. Your account has been adjusted.*	

a. If this reply is the agent's first participation in the interaction, in the reply, in the interaction thread, or in a reply in the interaction thread, the IRF for the agent includes an `ANCHOR_KEYS` value that indicates the applicable unique-participation metrics.

Populating Mediation Segments

The mediation segment fact (MSF) table describes interaction activity that involves mediation DNs, such as virtual and ACD queues, or multimedia interaction queues and workbins. The *grain* spans the time from when the interaction entered the mediation DN to the time that the interaction was abandoned in the mediation DN, cleared from the mediation DN (virtual queue only), or distributed from the mediation DN, including the time that it takes the interaction to be answered by the target resource or to be abandoned while alerting at the target resource. For voice, only completed ACD and virtual queue activity is populated; for multimedia interactions, both active and completed interaction queue, workbin, and virtual queue activity is populated.

Mediation Segment Fact Table

Genesys Info Mart stores mediation segment facts in the `MEDIATION_SEGMENT_FACT` (MSF) table. For detailed information about the columns in this table, refer to the *Genesys Info Mart Reference Manual* for your RDBMS.

Mediation Segments and Queues

An MSF record, or MSF, is created each time that an ACD or a virtual queue is used during interaction processing. An MSF might also be created each time that a multimedia interaction queue or a workbin is used during interaction processing, depending on configuration. For voice, mediation segments are populated in Genesys Info Mart only when the mediation segment is completed. For multimedia, both active and completed mediation segments are populated.

There are also links to the associated IRF, during which time the mediation that is represented by the MSF occurred.

Each MSF represents:

- The particular role played by the queue resource. For information about the resource roles that apply to queues, see Table 5 on page 52.
- The result of the association from the perspective of the queue resource to the target resource, as chosen during routing. For information about the technical results and technical result reasons that apply to MSFs for voice (ACD and virtual queues) and multimedia (interaction queue, workbin, or virtual queue), see Table 6 on page 55.

Configuration Options Used to Control Population of Queue Activity

The `[gim-etl-populate]` section of the Genesys Info Mart `Application` object contains options that enable or disable population of certain types of multimedia interaction activity in the MSF table. Some options can also be configured at script level, to override the application-level settings.

Note: ACD queue and virtual queue activity is always populated.

For more information about the `[gim-etl-populate]` options, see the configuration options reference chapter in the *Genesys Info Mart 8.1 Deployment Guide*.

Populating Mediation Segment Facts and Dimensions

Genesys Info Mart populates mediation segments in the following ways:

- The start time facts represent the start time of the mediation segment (when the interaction enters the queue).

  End time facts represent the end time of the mediation segment, which is one of the following:

  - The moment at which the interaction is abandoned while in the queue.
  - The moment at which the interaction is distributed from the queue to some target resource.
  - The moment at which the interaction is cleared from the queue, such as when a routing strategy routes the interaction from a parallel queue, or when it removes the interaction from the queue as it clears the routing targets for which it was waiting.

  For more information about how Genesys Info Mart represents dates, see Chapter 7 on page 165.
- The `TENANT` dimension identifies the tenant to which the queue resource belongs.
- The `RESOURCE_` dimension identifies the mediation DN resource that is associated with the mediation segment.
- The `TECHNICAL_DESCRIPTOR` dimension identifies the resource role and technical result of the mediation segment. For information about the resource roles and technical results that apply to mediation segments, see Chapter 3 on page 51.
- The `SHORT_ABANDONED_FLAG` indicates that, while waiting to be routed from the queue, the customer abandoned the interaction before the configured threshold expired. This enables these types of interactions to be filtered from the reports.
- The `MET_THRESHOLD_FLAG` indicates that the amount of time an interaction waited to be handled by a contact center resource was within a configurable threshold from the perspective of the queue. It is measured from the time that the interaction entered the queue to the time that it was answered by a contact center resource.
- The `ANSWER_THRESHOLD` contains the configured value used to calculate the `MET_THRESHOLD_FLAG` indicator.
- The `PLACE` dimension identifies the place that is associated with the target of the routing process.
- In addition to the mediation DN resource that is associated with the mediation segment, the `RESOURCE_` dimension identifies the contact center resource that was the routing target from the mediation DN.
- `MEDIATION_DURATION` is the length of time that the interaction was in the ACD queue, virtual queue, or interaction queue or workbin, based on timestamps from T-Server or Interaction Server.

In scenarios in which an interaction is bounced between a mediation resource and a strategy, as the strategy repeatedly retries busy agents, all the time that the interaction spends in a particular mediation resource is combined into a single MSF record, and the mediation duration includes all the interim strategy time—in other words, all strategy time except the time of the last strategy before the IRF.

In the case of an MSF for a virtual queue, a configurable option (`adjust-vq-time-by-strategy-time`) enables you to control whether the mediation duration includes or excludes time that the interaction spent in the strategy but outside the virtual queue.

- `ONLINE_DURATION` is the period of time that the interaction was in the ACD, virtual queue, interaction queue, or workbin before the interaction went offline.
- The `INTERACTION_TYPE` and `MEDIA_TYPE` dimensions are inherited from underlying IRF that has the lowest ordinal. This is the first resource fact that was created for the interaction and it generally has the earliest start time. In a network routing solution, all underlying network and premise resource facts are considered.
- The `RESOURCE_GROUP_COMBINATION` dimension records the virtual queue or queue membership in one or more groups.
- The `WORKBIN` dimension, if populated, indicates the workbin instance that is associated with the workbin mediation. This dimension enables downstream reporting applications to identify the type of resource and the specific resource that is associated with the workbin mediation.
- Provided that ICON provides the required information in the `G_ROUTE_RES_VQ_HIST` table, `IXN_RESOURCE_ID` links the MSF to an IRF that is considered to be the master record. In addition, `ENTRY_ORDINAL` indicates the order of entrance of this mediation segment relative to other mediation segments of the same IRF.

  These fields are populated for all MSF records, unlike `TARGET_IXN_RESOURCE_ID` (see below), which is populated in MSF records only for the devices that eventually distribute the interaction to a handling resource.

  These fields enable downstream reporting applications to provide detailed reports on mediation activity that was associated with a particular interaction or resource, even for interactions that were abandoned or cleared in virtual queues.
- `TARGET_IXN_RESOURCE_ID` provides a link between the MSF and the IRF that was the target of the routing process that is associated with the queue. This provides the means to associate the queue with the target of the routing strategy for virtual queue reporting.

  For voice interactions, a configuration option, `msf-target-route-thru-queue`, enables you to specify whether Genesys Info Mart considers the next handling resource or the party immediately following the Routing

Point to be the routing target, in scenarios in which a call is routed from a Routing Point through a virtual queue and then ACD queue to an agent ("route-thru-queue" scenarios).

- If the target is considered to be the next handling resource, the TARGET_IXN_RESOURCE_ID field in the MSFs for both the virtual queue and the ACD queue indicates the agent who ultimately answered the call. Furthermore, the technical result in the MSF for the virtual queue is AnsweredByAgent, and the MEDIATION_SEGMENT_ID and MEDIATION_RESOURCE_KEY fields of the associated IRF are set to the MEDIATION_SEGMENT_ID and RESOURCE_KEY of the MSF for the virtual queue.
- If the target is considered to be the party immediately following the Routing Point, TARGET_IXN_RESOURCE_ID is not populated in the MSF for the virtual queue, and the technical result is Diverted/Unspecified.

For more information about the option, see the description in the configuration options reference chapter in the *Genesys Info Mart 8.1 Deployment Guide*.

Populating Outbound Contact Campaign Activity

The Genesys Info Mart schema contains a number of subject areas related to Outbound Contact campaign activity. This section provides information about the Contact_Attempt subject area, which is the area that is focused on actual Outbound Contact campaign interactions.

Genesys Info Mart creates contact attempt facts in order to represent the attempts to reach the customer records of a calling list during the course of an Outbound Contact campaign.

Populating Contact Attempt Facts and Dimensions

Genesys Info Mart populates contact attempt facts as follows:

- The two references to the DATE_TIME dimension, in addition to the start and end timestamps, represent the start and end time, respectively, of the Outbound Contact attempt.
- For more information about how Genesys Info Mart represents dates and times of day, see Chapter 7 on page 165.
- The CAMPAIGN dimension identifies the Outbound Contact campaign that launched the attempt.
- The TENANT dimension identifies the tenant of the campaign.
- The GROUP_ dimension identifies the campaign group (agent group or place group) that is assigned to this campaign.

- The CALLING_LIST dimension identifies the calling list that contains the target record of the attempt.
- The RECORD_TYPE dimension identifies the type of the target record—for example, General or CampaignRescheduled.
- The RECORD_STATUS dimension identifies the status of the target record at the end of the contact attempt—for example, Updated or Cancelled.
- The CONTACT_INFO_TYPE dimension identifies the type of contact information that is provided in the target calling list record—for example, HomePhone or Mobile.
- The CALL_RESULT dimension is used to identify the final call result of the contact attempt (for example, Answer, Busy, or Wrong Party) as well as the dialer result (for example, Answer or Busy) if a dialer was used.
- The RESOURCE_ dimension identifies the resource that is associated with the first agent that corresponds to the Outbound Contact attempt, or an agent who is previewing this record.
- The RESOURCE_GROUP_COMBINATION_KEY dimension identifies the groups of which the Agent resource was a member when the contact attempt was started. This field references the default No Group value if the agent does not belong to a group.
- The PLACE dimension identifies the place that is associated with the first IVR DN or agent that corresponds to the Outbound Contact attempt.
- The DIALING_MODE dimension identifies the dialing mode that was used for the contact attempt—for example, Predictive, Progressive, or Preview. For GVP, these dialing modes are PROGRESSIVE_GVP, PREDICTIVE_GVP, and POWER_GVP, respectively.
- The MEDIA_TYPE dimension identifies the media type of the interaction that is associated with the Outbound Contact attempt—for example, Voice.
- The RECORD_FIELD_GROUP_1 and RECORD_FIELD_GROUP_2 dimensions contain custom fields from the calling list record. The values represent a snapshot that was taken at the end of the contact attempt.
- Record field facts in the CONTACT_ATTEMPT_FACT table hold custom field values from the target calling list record. The values represent the snapshot that was taken at the end of the contact attempt.
- State counts and durations summarize the amount of time that is spent on various activities.

Note: The following columns in the CONTACT_ATTEMPT_FACT table are no longer populated, although they remain in the schema:

- IXN_START_TIME
- IXN_START_TIME_KEY
- CONTACT_IXN_START_TIME
- CONTACT_WITHIN_DAILY_RANGE

To obtain the same data, use the following calculations:

- For IXN_START_TIME and CONTACT_IXN_START_TIME, make a join between CONTACT_ATTEMPT_FACT and INTERACTION_FACT on CONTACT_ATTEMPT_FACT.CALLID=INTERACTION_FACT.MEDIA_SERVER_IXN_GUID.
- For IXN_START_TIME_KEY, use INTERACTION_FACT.START_DATE_TIME_KEY.
- For CONTACT_WITHIN_DAILY_RANGE, you must also take into account the contact TIME_ZONE, which is identified by the TIME_ZONE_KEY. For assistance with this calculation, which is situation- and RDBMS-dependent, contact Genesys Customer Care.

Outbound Contact Campaign Activity Fact Tables

Genesys Info Mart stores facts about Outbound Contact campaigns and activity in the following tables:

- Contact attempts:
  - CONTACT_ATTEMPT_FACT
- Calling lists:
  - CALLING_LIST_METRIC_FACT
  - CALLING_LIST_TO_CAMP_FACT
- Campaigns and campaign groups:
  - CALLING_LIST_TO_CAMP_FACT
  - GROUP_TO_CAMPAIGN_FACT
  - CAMPAIGN_GROUP_SESSION_FACT
  - CAMPAIGN_GROUP_STATE_FACT

For detailed information about the columns in the Outbound Contact campaign fact tables, refer to the *Genesys Info Mart Reference Manual* for your RDBMS.

Populating Agent Activity Data

Genesys Agent activity data for both active and completed agent states is stored in summary tables for resource sessions, states, and reasons, for all media types. The summarized data, which is drawn from ICON, is stored in the following tables:

- SM_RES_SESSION_FACT
- SM_RES_STATE_FACT
- SM_RES_STATE_REASON_FACT

Do-Not-Disturb

Do-Not-Disturb (DND) status for each DN (or place and media type in the case of eServices/Multimedia) can optionally be factored into the

SM_RES_STATE_FACT and SM_RES_STATE_REASON_FACT tables, depending on the setting of the factor-dnd-into-sm-resource-states option. You configure this option in the [gim-etl] section on the Annex tab of the Switch configuration object.

DND is treated as a NOT_READY state, with the predefined software reason key DND On and no reason value. The termination of the DND state is treated as a READY state.

Agent State Hierarchy

Agent states are organized in a hierarchy, so that a higher-priority state takes precedence if multiple states happen simultaneously.

The default priority list (in descending order) is ACW, NOT_READY, BUSY, READY.

You can change the hierarchy of states in this table by adjusting the settings for the sm-resource-state-priority configuration option in the [gim-etl] section of the Info Mart Application object. However, be aware that, for parallel states, the state that is reported in the SM_RES_STATE_FACT and SM_RES_STATE_REASON_
FACT tables also depends on whether ICON has been set to interrupt after-call-work (ACW) and NotReady states when an agent places or receives another interaction (see "Obtaining Uninterrupted Voice AfterCallWork and NotReady Data").

Obtaining Uninterrupted Voice AfterCallWork and NotReady Data

Genesys Info Mart can represent voice ACW and NOT_READY states and reasons that are sourced from ICON and have them not be interrupted by incoming or outgoing calls that an agent makes while in these states.

To obtain uninterrupted ACW and NOT_READY data, set the gls-enable-acw-busy configuration option, which is located in the [gts] section on the Annex tab of the Switch configuration object, to 0 (the default setting is 1). This setting affects the agent model for parallel states.

Regardless of the configured priority list for parallel agent states in Genesys Info Mart, if gls-enable-acw-busy=0 and an agent goes into the ACW or NOT_READY state and then makes some calls on the same switch during ACW or NOT_READY, ICON considers such calls to be a part of the ACW or NOT_READY state.

This means that, even if the default priority list is changed to have BUSY take first precedence over ACW and NOT_READY, but ICON is configured not to interrupt ACW and NOT_READY states, the BUSY state is *not* recorded when it happens during uninterrupted ACW and NOT_READY states.

Populating Summarized Resource Sessions, States and Reasons

The SM_RES_SESSION_FACT, SM_RES_STATE_FACT, and SM_RES_STATE_REASON_FACT tables incorporate all data during the period in which an agent is logged on to a particular media type, regardless of the number of DNs or queues to which the agent logs on.

Starting with release 8.1, all three tables are populated based on the start and end time taken from corresponding IDB fields of the date format, rather than of the timestamp format that represents the data in seconds, in order to achieve a better precision in scenarios with very short agent states.

- For voice, the start and end times are taken in milliseconds, and this precision is used in internal calculations for summary voice agent states, reasons, and sessions. This ensures, for example, the proper alignment of multiple agent states that occur within the same second. The time values that result from calculations, including durations, are converted to a second format when stored in the Info Mart database.
- For multimedia, the start and end times are taken in seconds, which is the precision currently available from the data source. Although agent states, reasons, and sessions for agents handling multimedia interactions are calculated in milliseconds, the initial input has a one-second precision. The time values that result from calculations, including durations, are stored in the Info Mart database in seconds.

The SM_RES_SESSION_FACT Table

This table provides a summary of resource sessions by agent and media type.

In Genesys Info Mart release 8.1.0 and earlier, the populate-sm-resource-session-facts configuration option, which is in the [gim-etl-populate] section, controls whether the SM_RES_SESSION_FACT table is populated. Starting with release 8.1.1, because of performance improvements in SM_* transformation, Genesys Info Mart no longer needs to control population of the table, and hence Genesys Info Mart always populates the SM_RES_SESSION_FACT table.

The populate-sm-[media type]-resource-activity configuration options control which media types this table is populated with.

Note: GI2 reports require the SM_RES_SESSION_FACT table to be populated.

Each row of this table summarizes the login session(s) of all DNs and places that are associated with an agent relative to a given media type. The grain of the fact is an accumulating snapshot that represents the duration of the summary session.

A summary session represents the contiguous duration that an agent resource is logged on for a given media type, irrespective of the number of DNs and/or queues to which the agent resource logs on.

- For voice, a summary session starts when an agent resource first logs on to any voice DN-queue combination. The session continues, irrespective of how many other voice DNs and/or queues the agent logs on to. The session ends when the agent resource logs off all voice DNs and queues.
- For multimedia, a session is first created when the agent resource adds a media type to their login session or logs onto a DN that supports this media. The login session continues until the agent resource removes the media type from the last login session that includes this media type, or logs out of the last DN that includes this media type.

Start and end dates and times are stored as facts in the UTC time zone. Start and end date and times are also stored as a dimension reference for `DATE_TIME`. Both active and completed sessions are populated.

Note: In some multimedia scenarios, an agent can process interactions for a particular media type without logging into the media (that is, without adding the media type to a place). In this scenario, Genesys Reporting does not see agent states related to the processing of interactions for the media type that are not added to the agent's place. Therefore, to ensure correct reporting, Genesys recommends that agents take care to add a media to a place before handling interactions of this media type.

The SM_RES_STATE_FACT Table

Each row of this table describes a summarized agent resource state relative to a given media type. The grain of the fact is an accumulating snapshot that represents the duration of the summarized state.

A *summary state* represents the contiguous duration that an agent resource is logged on with a particular state for a given media type, irrespective of the number of DNs, places, and/or queues to which the agent resource logs on. The summary state is chosen from among the concurrent states of all DNs to which the agent is logged on, based on the configured state priority list. For multimedia, there are no DNs, so that the summarized state represents the state of the agent relative to the media type.

Do-Not-Disturb can optionally be factored into resource states in this table. This functionality is configurable by switch.

Note: GI2 reports require you to populate this table.

This table is sourced from Interaction Concentrator. The states that are recorded are the following:

- `Unknown` (the agent is logged on, but the agent state is unknown)

- Busy
- Ready
- NotReady
- AfterCallWork (voice media only)

Whether the NotReady or AfterCallWork (voice media only) states can be interrupted by interactions that the agent initiates or receives while in these states is dependent on the configuration of the underlying ICON application. The start and end dates and times are stored as facts in UTC time zone. The start date and time are also stored as dimension references for the DATE_TIME dimension.

The SM_RES_STATE_REASON_FACT Table

Each row of this table describes a summarized agent resource state reason and workmode relative to a given media type. The grain of the fact is an accumulating snapshot that represents the duration of the summarized state reason.

Note: You must set the Interaction Concentrator configuration option gls-active-reason-codes (in the [callconcentrator] section) to the mandatory value of TRUE. This ensures that the SM_RES_STATE_REASON_FACT table is consistent in situations in which the reason code state ends after the transformation of the interval in which this reason code started. If this option is *not* set to TRUE, the Genesys Info Mart configuration checker will log the problem and prevent any jobs from starting.

A *summary state reason* represents the contiguous duration for which an agent resource is in some state with a particular state reason for a given media type, irrespective of the number of DNs and/or queues to which the agent resource logs on. A reason code state that is written into this table should have a highest priority among all concurrent agent states. This means the same state (without reason) will occur in the SM_RES_STATE_FACT table.

When multiple reason codes occur simultaneously for one agent, Genesys Info Mart chooses one of them to record in the SM_RES_STATE_REASON_FACT table based on the following considerations:

- A software reason code takes priority over hardware.
- If the keys are different, the higher-value string takes priority.
- If the keys are the same, the key with the higher string value (not the higher numeric value) takes priority (using case-insensitive alphabetical comparison).
- The DND on reason takes the lowest priority with respect to other reason keys.

- Among two identical software reason codes with identical keys the priority is given to the state with the larger case-insensitive alphabetical reason code value.

Note: Reason code values are ranked alphabetically because ICON provides no data-type information to Genesys Info Mart that would identify whether the values are alphabetic, numerical, or mixed. As a result, some codes that occur in parallel may be ranked counterintuitively (`5 > 45`, for example).

Starting with release 8.1.4, a configuration option, `ignored-reason-codes` in the `[gim-transformation]` section, enables you to filter out reason codes that are not useful for reporting, so that they do not interfere with the priority rankings. Any hardware or software reason code keys specified by this option will not appear in the `RESOURCE_STATE_REASON` and `SM_RES_STATE_REASON_FACT` tables. For example, by default, Genesys Info Mart will ignore the `INTERACTION_WORKSPACE` key that Genesys License Reporting Manager (LRM) attaches to interactions to indicate that Genesys Workspace Desktop Edition (formerly known as Interaction Workspace [IWS]) is being used.

When a reason-code state has a lower priority than some other concurrent agent state *without* a reason, this reason code state is not recorded in the `SM_RES_STATE_REASON_FACT` table.

Detailed information on all of the simultaneous reason codes is available in the `GIDB_G_AGENT_STATE_RC_V`, `GIDB_G_AGENT_STATE_RC_MM`, `GIDB_G_AGENT_STATE_A_V`, and `GIDB_G_AGENT_STATE_A_MM` tables. Note that the `GIDB_G_AGENT_STATE_RC_V` and `GIDB_G_AGENT_STATE_RC_MM` tables may contain multiple records for a single interaction, differing in their ending timestamp, if a reason-code state starts in one extract interval and ends in another extract interval.

Note: Do-Not-Disturb is optionally factored into summary state reasons with the predefined reason code key `DND On` and no reason value, based on the configuration of the underlying Switch object. All reasons that are associated with the current highest priority state of the agent are recorded. Genesys Interactive Insights reports require you to populate this table.

This table is sourced from IDB. The states for which reasons are recorded are the following:

- `Ready`
- `NotReady`
- `AfterCallWork` (voice media only)

Starting with release 8.0, Genesys Info Mart does not provide a default reason for the `NotReady` state.

Whether the NotReady or AfterCallWork (voice media only) state can be interrupted by interactions that the agent initiates or receives while in these states is dependent on the configuration of the underlying ICON application.

The start and end dates and times are stored as facts in the UTC time zone. The start date and time are also stored as dimension references for the DATE_TIME dimension.

How Summarized Data Is Processed

Summarized agent data, which must be recorded by a single ICON instance for a given agent, is processed in the following ways:

- Genesys Info Mart combines information for the same agent and media type from the ICON GX_SESSION_ENDPOINT table to form summarized media type sessions.
  - For voice, Genesys Info Mart combines information for the same agent and media type from the G_AGENT_STATE_HISTORY, G_AGENT_STATE_RC, GIDB_G_AGENT_STATE_A, and G_DND_HISTORY tables in IDB to create summarized states and reasons that optionally have Do-Not-Disturb status factored into them. In addition, if the agent is logged on to more than one voice DN at a time, a configurable state priority list is used to determine which DN's state is considered to be the winning state.
  - For multimedia, Genesys Info Mart combines information for the same agent and media type from the G_AGENT_STATE_HISTORY, G_AGENT_STATE_RC, GIDB_G_AGENT_STATE_A, G_DND_HISTORY, and GX_SESSION_ENDPOINT tables in IDB to form summarized states and reasons that optionally have Do-Not-Disturb status factored into them.

Special Considerations For Very Short-Duration States

The three summary tables have one-second data granularity. Let's consider what this means when a summary is calculated for agent states, reasons, or sessions with very short durations.

Let's use agent states and the SM_RES_STATE_FACT table as an example. When two or more states happen within the same second, the processing differs slightly between voice and multimedia data.

- For voice, the SM_RES_STATE_FACT_KEY determines the state sequence, which is set based on the earliest start time, in milliseconds, of the overlapping states. Lower-priority states with durations of less than one second are still taken into calculation, which is performed with millisecond precision.
- For multimedia, the SM_RES_STATE_FACT_KEY sequence key is set based on the earliest start time of the overlapping states in seconds. Unlike with voice, the input for the states for agents handling multimedia interactions has one-second precision; the calculations to produce summary states are made in milliseconds. Only the highest-priority state is recorded for the

duration of the same second. As a result, some lower-priority states with durations of less than one second may be hidden by a higher-priority state and disappear from the stored data.

For example, the following scenario is possible for an agent who handles multimedia interactions:

An agent goes from the `NotReady` state to the `Ready` state and, being ready, receives an interaction within a fraction of a second. Then, agent goes into the `Busy` state. Since the `Ready` state, by default, has lower priority, it may disappear completely if its duration is less than one second. As a result, the IRF table may display the previous summarized state of the agent as `NotReady`, although it was actually `Ready`.

If necessary, you can obtain the exact agent state on a specific DN just before the agent takes an interaction from the GIDB tables.

The processing logic for very short reasons and sessions is similar to the logic described for very short agent states.

Note: The timestamps for the start and end time in summarized tables may not match times in the IDB tables. `END_TS` in summarized tables means the beginning of the second by which the state has ended.

The difference in stored times becomes greater in complex scenarios with multiple simultaneous states for the same agent.

Moreover, the state sequence order may be incorrect in deployments with multiple Interaction Concentrator instances, because of time synchronization errors between the hosts.

Special Considerations for Long-Duration Sessions or States

Given usual contact-center organization and policies, Genesys reporting does not expect agent login sessions or states to be very long-lasting. However, in practice, agent sessions and states might last indefinitely—for example, if agents never log out.

From the point of view of Genesys Info Mart operations, long-lasting agent sessions and states negatively affect transformation performance. From the point of view of data quality, very long-lasting agent sessions or states can yield misleading reporting results—for example, if shift reporting (perhaps used for agent compensation) is based on unrealistic agent-activity data.

For these reasons, Genesys Info Mart provides functionality to apply timeouts to agent login sessions and states that exceed configurable maximum durations. By default, Genesys Info Mart allows a maximum duration of 24 hours for login sessions and, starting with release 8.1.3, 4 hours for each instance of an agent state within a login session. You can change the respective maximum durations by adjusting the settings for the `max-session-duration-in-hours` and `max-state-duration` options in the `[gim-etl]` section of the Genesys Info Mart `Application` object. (The `max-state-duration` option was

introduced in release 8.1.3.) For more information about the configuration options, see the *Genesys Info Mart 8.1 Deployment Guide*.

Detecting Session Inactivity

The timeout implementation enables Genesys Info Mart to detect when a session has gone inactive.

Starting with release 8.1.3, Genesys Info Mart will end the session when all states have ended, even if the end of the session has not been extracted and the session has not yet timed out. For example, if a state is timed out by `max-state-duration` and there are no other active states, then Genesys Info Mart deems the session to be inactive and terminates it.

Recognition of sessions that have gone inactive can provide more useful reporting on situations in which agents forget to log out. The smaller the value of `max-state-duration`, the sooner Genesys Info Mart will detect the session inactivity.

Handling Resumed Session Activity

Starting with release 8.1.3, if a state transition occurs in a session that Genesys Info Mart previously timed out or ended because of inactivity, Genesys Info Mart creates a new session beginning with the new state. The new session continues until the first of the following occurs:

- All states in the new session have ended or have timed out.
- The new session times out after `max-session-duration-in-hours`.

Prior to release 8.1.3, if states occur after a session times out, Genesys Info Mart does not associate these states with a login session (`SM_RES_STATE_FACT.SM_RES_SESSION_FACT_KEY = -1`).

Note: After a state has been timed out by `max-state-duration`, if there is a new resource state reason for that state, the reason will not be associated with any state or session:

- `SM_RES_STATE_REASON_FACT.SM_RES_SESSION_FACT_KEY = -1`
- `SM_RES_STATE_REASON_FACT.SM_RES_STATE_FACT_KEY = -1`

Special Case with No Contact Center Activity

In the rare event that there is no call or interaction activity in the contact center, agent states are updated only after some delay. You can minimize this delay by setting an appropriate value for an ICON configuration option, `dss-no-data-tout`. The default value is `300` seconds. As a result, by default there is a five-minute (300 second) delay before Info Mart sees that the agents have no interaction states. Genesys recommends that you reduce the delay to 60 seconds.

Populating Do-Not-Disturb Data

Do-Not-Disturb data is optionally factored into states and reasons in the summarized `SM_RES_STATE_FACT` and `SM_RES_STATE_REASON_FACT` tables for all media types.

Including Do-Not-Disturb Data in Summary Tables

Inclusion of Do-Not-Disturb data in the summarized SM_RES_STATE_FACT and SM_RES_STATE_REASON_FACT tables is controlled by the factor-dnd-into-sm-resource-states configuration option, which is located in the [gim-etl] section under the Annex tab of each switch. The default setting is TRUE for eServices/Multimedia switches and FALSE for voice switches.

For eServices/Multimedia, Do-Not-Disturb is treated as a global NotReady for all media types to which an agent is logged on at a given place.

DND states are treated as NotReady with a reason that indicates DND on. An example, how DND state is calculated for the default state priority list (AfterCallWork, NOT_READY, BUSY, READY, UNKNOWN) is explained in Table 4 on page 48. This logic might be different for a user-configured state priority list.

Users can configure state priority in the sm-resource-state-priority option, in the [gim-etl-populate] section of the Genesys Info Mart Application object.

Table 4: Calculating DND Status

Conditions	Resulting DND Status
DND is turned *on* and the declared state is currently Ready.	The resource is considered to be in a NotReady state with a reason that indicates DND On.
DND is turned *off* and the declared state was previously Ready.	The resource returns to Ready with whatever reasons were originally attached to the Ready request.
DND is turned *on* and the declared state is currently AfterCallWork.	The resource stays in the AfterCallWork state. If AfterCallWork ends before DND is turned back off, the resource becomes NotReady, and the reason is DND On. If DND is turned on and off during AfterCallWork, the resource state is never shown as NotReady. **Note:** AfterCallWork applies only to non-multimedia media types.

Table 4: Calculating DND Status (Continued)

Conditions	Resulting DND Status
The resource is in `NotReady` state and DND is turned *on* or *off*.	Any `NotReady` reasons that are currently in effect are not interrupted. If an existing `NotReady` state had no reasons, a new `NotReady` reason state with the key `DND On` is added.
The resource is in `Busy` state and DND is turned *on*.	The resource immediately enters the `NotReady` state with `DND On` as the reason, and the `Busy` state is closed.

Chapter

3 Technical Descriptors

Understanding when interaction resource facts (IRFs) and mediation segment facts (MSFs) are created can help you to determine which types of interaction resources and mediation segments to include in, or exclude from, your queries. The `TECHNICAL_DESCIRIPTOR` dimension is a combination of attributes—resource roles, role reasons, technical results, and technical result reasons—that describe how interactions arrive at and depart from resources.

This chapter describes the technical descriptor combinations that are applicable for IRFs and MSFs for voice and multimedia. It contains the following sections:

After you understand the resource roles and technical results, see Chapter 4 on page 65 for diagrams that depict the IRFs that result from typical voice interaction flows and Chapter 5 on page 125 for diagrams that depict the IRFs that result from typical multimedia interaction flows.

Resources and Resource Roles

Some technical descriptors apply only to IRFs, some apply only to MSFs, and some apply to both. Similarly, some technical descriptors apply only to voice interactions, some apply only to multimedia interactions, and some apply to both.

Whether a particular resource role or technical result applies to IRFs or MSFs for various media types depends on the type of resource.

Resource Roles

The resource role of the interaction-handling or mediation resource depends on how the interaction arrives at the resource.

The number of potential resource roles for mediation resources is limited. For an ACD queue, virtual queue, multimedia interaction queue, or multimedia workbin, each row in the MSF table has a `resource_role` of `Received` or `Received_Consult`.

Summary of Resource Roles

Table 5 describes the resource roles and role reasons that Genesys Info Mart uses for resources that are the subjects of IRF or MSF records. Except where specified otherwise, all resource roles apply to both voice and multimedia interactions.

For a list of the available combinations of resource roles with the other technical-descriptor attributes, see the Appendix on page 177.

Note: The `TECHNICAL_DESCRIPTOR` dimension table includes some combinations of resource roles and role reasons that Genesys Info Mart does not use. Table 5 does not describe these combinations.

Table 5: Resource Roles and Role Reasons

Resource Role	Description	Comment
DivertedTo	Denotes an interaction that was delivered to the resource via an ACD queue.	Applies to: IRF (voice). A resource role of `DivertedTo` paired with a technical result of `Conferenced` indicates the initiator of a conference call.
InConference	Denotes that the IRF was created for a resource as the result of a conference call in which the resource joined the conference.	Applies to: IRF. A resource role of `InConference` paired with a technical result of `Conferenced` indicates that, after joining the conference, the joining resource was the initiator of a subsequent conference.
Initiated	Denotes that the resource in the IRF row initiated either an internal interaction or an outbound interaction.	Applies to: IRF. A resource role of `Initiated` paired with a technical result of `Conferenced` indicates that the resource initiated a call and was the initiator of a conference call.

Table 5: Resource Roles and Role Reasons (Continued)

Resource Role	Description	Comment
InitiatedConsult	In the separate IRF record that is created when an agent or IVR initiates a consultation, denotes that the subject of the IRF initiated the consultation.	Applies to: IRF. This resource role indicates that the subject of the IRF initiated a consultation, mute transfer, two-step transfer, or two-step conference to another resource.
Puller	Denotes that the resource pulled the multimedia interaction from an Interaction Queue or Interaction Workbin.	Applies to: IRF (multimedia).
Received	For IRFs, denotes that the resource received an inbound interaction without the benefit of prior distribution devices moving the call to it. This is typical for internal call types that are dialed directly to the resource. For MSFs, this resource role applies to all nonconsultation interactions that are received into a queue.	Applies to: IRF and MSF. A resource role of `Received` paired with a technical result of `Conferenced` in the IRF context indicates the initiator of a conference call.
ReceivedConsult	Denotes that the IRF or MSF was created for a resource as the result of a consultation only (the resource did not receive a transfer, or was not joined into a conference). This enables counting of consultations that are received by a resource.	Applies to: IRF (voice and multimedia) and MSF (voice). • A resource role of `ReceivedConsult` paired with a technical result of `Conferenced` represents the unlikely event that a resource receives a consultation, consults another resource, and then creates a conference call between the resources. This combination in the IRF context indicates the initiator of a conference call. • For MSF records, this resource role indicates that the interaction arrived in the mediation resource as the result of a consultation between contact center resources and was still in consultation when the interaction was diverted by the mediation resource.

Table 5: Resource Roles and Role Reasons (Continued)

Resource Role	Description	Comment
ReceivedTransfer	Denotes that the IRF was created as a result of the interaction being transferred to the IRF resource by a resource other than a nonself-service IVR, either directly or indirectly through an intermediate redirecting resource.	Applies to: IRF. A resource role of `ReceivedTransfer` paired with a technical result of `Conferenced` indicates the initiator of a conference call.
RedirectedTo	An interaction has been returned to the queue from which it was pulled. **Note:** An IRF is created for a queue only if the interaction ended in the queue—for example, if the technical result was `CustomerAbandoned`.	Applies to: IRF (multimedia). An interaction is redirected back to a queue if: • A routing strategy pulled the interaction from the queue and offered it to an agent, but the agent did not accept the invitation into the interaction. • The interaction has been assigned to an agent for longer than the handling timeout that is configured in Interaction Server. • The interaction has been assigned to a routing strategy for longer than the routing timeout that is configured in Interaction Server.
RoutedTo	Denotes an interaction that was delivered to the resource via a routing point.	Applies to: IRF. For voice interactions, a resource role of `RoutedTo` paired with a technical result of `Conferenced` indicates the initiator of a conference call.
Unknown	Genesys Info Mart does not have sufficient information to determine the resource role.	Applies to: IRF and MSF.

Technical Results

The technical result and technical result reason of the IRF or MSF depend on how the interaction leaves the resource.

Summary of Technical Results

Table 6 describes the technical results and technical result reasons that Genesys Info Mart uses for resources that are the subjects of IRF or MSF records. Except where specified otherwise, all technical results and technical result reasons apply to both voice and multimedia interactions.

For a list of the available combinations of technical results with the other technical-descriptor attributes, see the Appendix on page 177.

Note: The TECHNICAL_DESCRIPTOR dimension table includes some combinations of technical results and technical result reasons that Genesys Info Mart does not use. Table 6 does not describe these combinations.

Table 6: Technical Results and Technical Result Reasons

Technical Result	Description	
	Result Reason	Comment
Abandoned	Denotes that processing of the interaction by the resource did not complete normally.	
	AbandonedWhileQueued	Applies to: IRF and MSF. The interaction was abandoned while in the queue.
	Redirected	Applies to: IRF (voice) and MSF (voice). In the IRF context, processing of the voice interaction by the resource that is the subject of the IRF row was abandoned, and the interaction was redirected to another resource. In the MSF context, processing of the voice interaction by a target handling resource was abandoned, and the interaction was redirected to another resource.
	Rejected	Applies to: IRF (multimedia) and MSF (multimedia). A handling resource, which was an agent (or a place), was invited into the interaction but rejected the invitation. As a result, processing of the interaction was abandoned. In the IRF context, the resource that rejected the invitation is the subject of the IRF record. In the MSF context, the resource that rejected the invitation is a target handling resource.

Table 6: Technical Results and Technical Result Reasons (Continued)

Technical Result	Description	
	Result Reason	**Comment**
Abandoned (cont.)	Revoked	Applies to: IRF (multimedia) and MSF (multimedia). A handling resource, which was an agent (or a place), was invited into the interaction, but the invitation was revoked when the resource did not accept the invitation before the `handling-timeout` that is configured in Interaction Server. As a result, processing of the interaction was abandoned. In the IRF context, the resource that did not accept the invitation in time is the subject of the IRF record. In the MSF context, the resource that did not accept the invitation in time is a target handling resource.
	Unspecified	Applies to: IRF and MSF. In MSF records with a resource role of `ReceivedConsult`: • For virtual queues, either the consultation was abandoned or a consultation was retrieved while in the virtual queue. • For ACD queues, the consultation mediation attempt through this ACD queue was abandoned or retrieved while waiting for service.
AbnormalStop	(Multimedia only) Denotes that the interaction was stopped by an entity (for example, Interaction Server or a Media Server) that was not a party to the interaction, in situations in which no other technical result applies. For example, `AbnormalStop` would not apply if the Media Server stops the interaction with a reason system name of `Abandoned`, because the technical result of `CustomerAbandoned` would apply to that scenario. The `STOP_ACTION` field in the last IRF for the interaction is 0.	
	AbnormalStopWhile Queued	Applies to: IRF and MSF The interaction was stopped while in the virtual queue, interaction queue, or workbin.
	AbnormalStopWhile Ringing	Applies to: IRF
	Unspecified	Applies to: IRF and MSF

Table 6: Technical Results and Technical Result Reasons (Continued)

Technical Result	Description	
	Result Reason	**Comment**
CustomerAbandoned	Denotes that the customer initiated termination of the interaction. **Note:** In order for Genesys Info Mart to report that a multimedia interaction has been abandoned by the customer, the Media Server must operate in compatibility mode (with the Chat Server `stop-abandoned-interaction` configuration option set to `true`). For the reason for this requirement, see "Abandoned" on page 63.	
	AbandonedFromHold	Applies to: IRF (voice). The handling resource placed the interaction on hold, and the customer abandoned the interaction.
	AbandonedWhileQueued	Applies to: IRF and MSF. In MSF records, this technical result combination indicates that: • For virtual queues, interaction queues, or workbins, the interaction was abandoned while in the mediation resource. • For ACD queues, the mediation attempt through this ACD queue was abandoned while waiting for service.
	AbandonedWhileRinging	Applies to: IRF.
	Unspecified	Applies to: IRF and MSF.
Diverted	Denotes that the mediation resource diverted the interaction to a target resource. Applies to: MSF.	
	AbandonedWhileRinging	The interaction was abandoned before the target resource could answer it. For voice interactions, the target was a handling resource (Agent, IVR or ACD position DN) that had a talk count = 0, and re-route on no answer (RONA) did not occur.
	AbnormalStopWhileRinging	(Multimedia only) Before the target resource answered, the interaction was stopped by an entity that was not a party to the interaction (for example, by a Media Server).
	AnsweredByAgent	The target resource was an agent, and the agent answered the interaction. For voice interactions, the target resource was an agent who had a talk count > 0.

Table 6: Technical Results and Technical Result Reasons (Continued)

Technical Result	Description	
	Result Reason	**Comment**
Diverted (cont.)	AnsweredByOther	The target resource was not an agent, and it answered the interaction. For multimedia interactions, the target resource was a place, but no agent was logged in to that place. For voice interactions, the target was a resource, other than an agent, that had a talk count > 0 (typically an IVR or ACD Position DN).
	Redirected	The target resource did not answer the interaction; as a result, the interaction was routed to another resource. For voice interactions, the target was a resource that was re-routed on no answer (RONA'd) or that forwarded the interaction elsewhere.
	Rejected	(Multimedia only) The target resource was an agent (or a place). The agent (or place) was invited into the interaction, but the invitation was rejected. As a result, the interaction is placed back into the interaction queue from which it came.
	Revoked	(Multimedia only) The target resource was an agent (or a place) that was invited into the interaction, but the invitation was not accepted before the `delivering-timeout` that was configured in Interaction Server. As a result, the interaction was placed back into the interaction queue from which it came.
	RoutedToOther	The target was a mediation resource that was not the subject of the IRF.
	RouteOnNoAnswer	(Voice only) The target resource was an agent; the call rang at the handling resource, was not answered, and was deflected to another resource.
	Unspecified	

Table 6: Technical Results and Technical Result Reasons (Continued)

Technical Result	Description	
	Result Reason	**Comment**
Cleared	Denotes that the interaction was cleared from a queue. Applies to: MSF.	
	DefaultRoutedByStrategy	(Virtual queues only) The interaction was routed by Universal Routing Server (URS) to the default destination, as defined by the URS configuration options.
	DefaultRoutedBySwitch	(Voice virtual queues only) The switch default-routed the interaction.
	PulledBack (starting with release 8.1.4) or PulledBackTimeout (in releases earlier than 8.1.4)	(Multimedia only) The routing strategy was unable to route the interaction successfully before the expiration of the `routing-timeout` that was configured in Interaction Server. As a result, the routing was considered to be a failure and the interaction was taken from the routing strategy and placed back into the interaction queue from which it came.
	RoutedFromAnotherVQ	(Virtual queues only) The interaction was added to this virtual queue as well as to a parallel virtual queue. It was routed from the parallel virtual queue to the target destination, and it was cleared from this virtual queue.
	Stopped	(Multimedia only) The interaction was stopped while in mediation, in situations in which neither `CustomerAbandoned` nor `AbnormalStop` applies.
	StuckCall	(Virtual queues only) An interaction that Interaction Concentrator (ICON) identified as a stuck call was cleared from the virtual queue. (ICON determines that an interaction is stuck in a virtual queue if ICON received an event that indicates that the interaction entered the virtual queue, but ICON did not receive the event that indicates that the interaction exited the virtual queue, and URS has stopped sending status updates for that interaction.) **Note:** To calculate durations from virtual queue data accurately, Genesys recommends that rows that have this technical result and reason not be used.

Table 6: Technical Results and Technical Result Reasons (Continued)

Technical Result	Description	
	Result Reason	**Comment**
Cleared (cont.)	Targets Cleared	(Virtual queues only) The interaction was cleared from the virtual queue by the URS strategy `Clear Target` function.
	Unspecified	For virtual queues, usually indicates that the interaction was cleared from the virtual queue because no target was found. For ACD queues, usually indicates that the interaction was parallel queued and was not diverted from this ACD queue to another contact center resource.
Completed	Denotes that processing of the interaction by the resource completed normally. Applies to: IRF.	
	Archived	(Multimedia only) The interaction was placed into an Interaction Queue that, based on the value of the `completed-queues` configuration option, Genesys Info Mart identifies as an archive queue for completed interactions. This Result Reason was introduced in release 8.1.3, to improve reporting in Genesys intelligent Workload Distribution (iWD) or other scenarios in which interactions are placed into “archiving” queues, instead of being terminated immediately after processing.
	Canceled	(Multimedia only) The interaction was placed into an Interaction Queue that, based on the value of the `canceled-queues` configuration option, Genesys Info Mart identifies as an archive queue for canceled interactions. This Result Reason was introduced in release 8.1.3, to improve reporting in iWD or other scenarios in which interactions are placed into “archiving” queues, instead of being terminated immediately after processing.
	Unspecified	

Table 6: Technical Results and Technical Result Reasons (Continued)

Technical Result	Description	
	Result Reason	**Comment**
Conferenced	Denotes that the interaction resulted in a conference. See comments in Table 5 on page 52 for the meaning of specific combinations of the `Conferenced` technical result with various resource roles. Applies to: IRF (voice).	
	Unspecified	
DestinationBusy	Denotes that the interaction did not reach the target resource because the destination was busy. Applies to: IRF (voice)	
	Unspecified	
OutboundStopped	An outbound interaction was created and stopped without being sent. Applies to: IRF (multimedia)	
	Unspecified	
Pulled	Denotes that the interaction was pulled from an Interaction Queue or Interaction Workbin. Applies to MSF (Multimedia).	
	Unspecified	

Table 6: Technical Results and Technical Result Reasons (Continued)

Technical Result	Description	
	Result Reason	**Comment**
Redirected	Denotes that an interaction was redirected to another resource. Applies to: IRF.	
	PulledBack (starting with release 8.1.4) or PulledBackTimeout (in releases earlier than 8.1.4)	(Multimedia only) The agent did not handle the interaction before the `handling-timeout` that is configured in Interaction Server. As a result, the interaction was placed back into the interaction queue from which it came. Starting with release 8.1.4, when paired with a resource role of `InConference`, identifies the uncommon scenario in which an agent who was invited into a chat conference became the only remaining agent in the chat (in other words, the inviting agent left the chat), and then the remaining agent left the chat abnormally (for example, because the agent logged out while the interaction was still open, or the agent's desktop application terminated unexpectedly while the interaction was still open). As a result, Interaction Server pulled the interaction back from the agent and placed the interaction in a queue.
	RouteOnNoAnswer	(Voice only) The interaction was diverted from an agent or IVR to another contact center resource as the result of a ring no answer.
	Unspecified	For voice interactions, the interaction was diverted from an agent when forwarded to another resource, such as voice mail.
Transferred	Denotes that the resource completed a transfer of the interaction to another resource. Applies to: IRF.	
	Unspecified	

Multimedia Stop Reason System Names

One of the reporting event attributes captured by eServices Interaction Server is the reason system name (`attr_reason_system_name`) associated with a request. The reason system name associated with a Stop Processing request is of particular significance and is captured by ICON in the `G_STOP_REASON` column of the `GM_L_USERDATA` table. There are certain Stop Processing reason

names which are meaningful to Genesys Info Mart to correctly report the Technical Result:

Abandoned

In Media Server compatibility mode (described in the information on the Chat Server configuration option, `stop-abandoned-interaction`, in the eServices/Multimedia documentation), a chat interaction is stopped with a reason system name of `Abandoned` when it is abandoned by the customer. Genesys Info Mart uses this stop reason to determine if a chat interaction has been abandoned.

Sent

When an outbound-sending e-mail strategy sends an e-mail outside of the contact center, by convention, as illustrated in the Interaction Workflow Samples, the Strategy stops the outbound e-mail interaction with a reason system name of `Sent`. Genesys Info Mart relies upon this convention, and uses this stop reason to determine if an outbound e-mail was actually sent.

Normal

A stop reason of `Normal` may be used in a large variety of contexts, but there is only one scenario where its use affects Genesys Info Mart processing.

When an agent transfers a chat interaction to a Chat Transcript Queue, a Chat Transcript Strategy pulls the interaction from the queue, and decides whether or not to send an e-mail transcript of the chat interaction, based upon user data attached by Genesys Agent Desktop, and, by convention, as illustrated in the Interaction Workflow Samples, stops the chat interaction with a reason system name of `Normal`. Genesys Info Mart relies upon this convention to determine how to represent the action of the agent that transferred the chat interaction to the Chat Transcript Queue. In this case, the agent, who transferred the interaction to the Chat Transcript Queue, is not attempting to *transfer* the interaction to another resource, but instead has *completed* the chat activity, and the transfer action is to engage follow-up workflow processing.

Network Call Scenarios

In certain network call scenarios, Genesys Info Mart 8.x reporting results differ from Genesys Info Mart 7.6 results. In particular, in network call flows in which:

- The call is sent back to the Network switch at least once before the call reaches a handling resource, the IRF row for the handling resource no longer contains an incorrect resource role of `ReceivedTransfer`.

- The call is pulled back to the Network switch while the call is connected at a handling resource, the IRF row for this resource will have a technical result of `Completed` instead of `Transferred`.

Chapter

4 Validated Voice Interaction Flows

This chapter describes the recognized, validated voice interactions that have been tested and that are supported by Genesys Info Mart. The validated interactions are premise-based flows that involve one or more of the deployed Genesys solutions.

The call flows that are described in this chapter are intended as examples that you can modify for your environment. However, Genesys does not guarantee results for modified interaction flows.

This chapter contains the following sections:

Note: Voice interactions that are generated by other supported Genesys solutions might yield call flows in Genesys Info Mart that do not directly translate to the call flows that are described in this chapter. Voice interactions that involve Genesys solutions and are not supported by Genesys Info Mart might yield unpredictable results.

Overview

The validated call flows that are described in this chapter are organized according to the types of solution that might be deployed in your contact center. Table 7 provides an overview of the validated call flows.

Table 7: Validated Call Flows

Solution	Description
Framework only	Based on the dialed number, voice interactions that arrive at the switch are queued to an ACD queue that represents a requested skill, service type, or customer segment. Agents who are logged into the ACD queues handle the interactions. **Note:** Flows that start in a diagram under one of the other solutions can resume in another diagram under this solution (for example, if a voice interaction in Universal Routing is routed to an agent, and the agent performs a two-step transfer to another agent).
IVR in front of switch	Voice interactions arrive at an IVR that is visible to the IVR Server's virtual T-Server. The focus of the IVR application can be either self-service or simple front-end identification and segmentation. If the IVR application cannot completely handle the voice interaction, the interaction can be transferred to an ACD queue behind the switch that represents a requested skill, service type, or customer segment. Agents logged in to the ACD queues handle the interactions.
IVR behind switch	Voice interactions that arrive at the switch are queued to an ACD queue, where the ACD positions are actually IVR DNs. The focus of the IVR application can be either self-service or simple front-end identification and segmentation. If the IVR application cannot completely handle the voice interaction, the interaction can be transferred to an ACD queue that represents a requested skill, service type, or customer segment. Agents who are logged in to the ACD queues handle the interactions.
Universal Routing	Voice interactions that arrive at the switch are delivered to a Routing Point. Universal Routing Server (URS) uses criteria such as ANI, DNIS, and the date and time of day to collect information and select an appropriate routing target. Basic targets are ACD queues and individual DNs; more advanced targets are agent groups, place groups, and skill expressions.
Universal Routing assisted by IVR behind switch	Voice interactions that arrive at the switch are queued to an ACD queue, where the ACD positions are actually IVR DNs. The IVR application collects digits and information about the caller, and transfers the call to a Routing Point. Universal Routing uses the collected information to select an appropriate routing target. Basic targets are ACD queues and individual DNs. More advanced targets are agent groups, place groups, and skill expressions.

Table 7: Validated Call Flows (Continued)

Solution	Description
IVR in front of switch assisted by Universal Routing	Voice interactions arrive at an IVR that is visible to the IVR Server's virtual T-Server. Through a Routing Point in the IVR Server's virtual T-Server, the IVR application invokes a Universal Routing strategy. Universal Routing instructs the IVR application to play applications or collect information. Universal Routing uses the collected information to return an appropriate target. The IVR application hook-flash transfers the call to that target.
IVR behind switch assisted by Universal Routing	Voice interactions that arrive at the switch are queued to an ACD queue, where the ACD positions are actually IVR DNs. Through a virtual routing point in the premise T-Server, the IVR application invokes a Universal Routing strategy. Universal Routing instructs the IVR application to play applications or collect information. Universal Routing uses the collected information to return an appropriate target. The IVR application mute transfers the call to that target.

Diagram Conventions

The call flow diagrams in this chapter use the following conventions:

- Dotted shading indicates customer wait time.
- Diagonal shading indicates customer handle time.
- The following abbreviations are used for simplicity:
  - IRF—Interaction Resource Fact
  - SS IVR—Self-service IVR (considered to be a *handling resource* or *resource of interest* with regard to IRF data collection)
  - nonSS IVR—Nonself-service IVR (considered *not* to be a *handling resource* or *resource of interest* with regard to IRF data collection)

Figure 2 on page 68 shows a legend for the call flow diagrams.

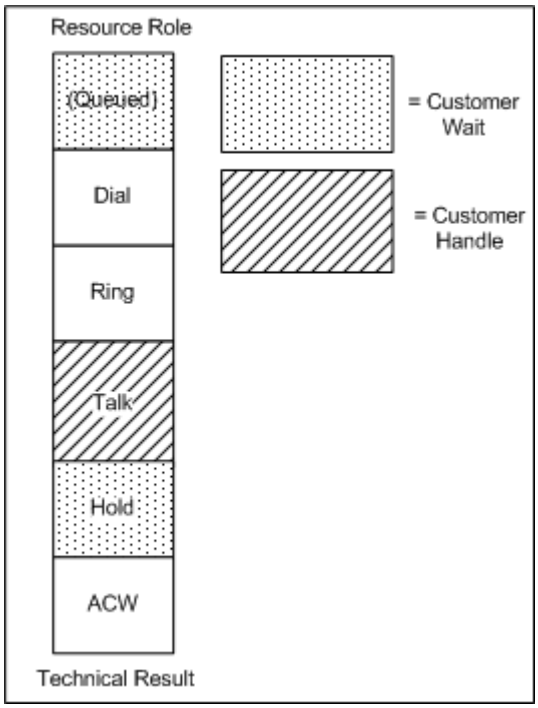

Figure 2: Call Flow Legend

To show the voice interaction flow, the diagrams in this chapter depict the media-specific states in sequence.

Figure 3 shows an example of an IRF call flow.

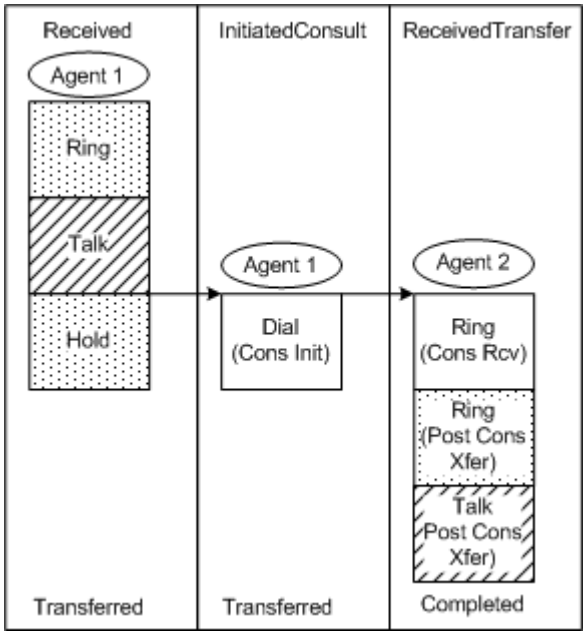

Figure 3: Sample Call Flow

Notes on the Interaction-Flow Diagrams

The diagrams represent the resources that participate in the interaction and their states.

The following list points out features that are specific to the diagrams:

- The circled resource in the diagram represents the resource that is the subject of the IRF record (the *resource of interest*).
- The vertical lines indicate separate IRF rows.
- The resources of interest are *handling resources*, which are the resources that have the greatest interest for reporting—agents, self-service IVRs, and DNs without an agent. *Nonhandling resources* include mediation resources such as queues, routing points, and nonself-service IVRs.
- The diagrams also show with which portion of the call each resource's state is associated (such as received consult, post-consult transfer, and post-consult conference). The resource role is shown above the IRF and the technical result below it.

Framework-Only Call Flows

Based on the dialed number, voice interactions that arrive at the switch are queued to an ACD queue that represents a requested skill, service type, or customer segment. Agents who are logged in to the ACD queues handle the interactions.

Note: Flows that start under a diagram in one of the other solutions might resume in another diagram under this solution (such as when a voice interaction in Universal Routing routes to an agent, and the agent performs a two-step transfer to another agent).

This section describes call flows for the following types of interactions:

- Inbound call flow examples
- Outbound call flow example (see page 87)
- Internal call flow examples (see page 88)

Inbound Call Flow Examples

This subsection contains several examples of inbound call flows. Each example represents a different outcome:

- An ACD queue directs the inbound call to an agent.
- The inbound call is answered directly by an agent (see page 71).
- An agent mute transfers the call to an ACD queue (see page 71).
- An agent mute transfers the call to another agent (see page 74).

- An agent consults to an ACD queue and then retrieves the call (see page 75).
- An agent consults to another agent and then retrieves the call (see page 78).
- An agent consults to an ACD queue and then transfers the call (see page 79).
- An agent consults to another agent and then transfers the call (see page 80).
- An agent consults to an ACD queue and then conferences the call (see page 81).
- An agent consults to another agent and then conferences the call (see page 82).
- An agent consults to another agent and then conferences the call. Subsequently, the second agent consults to a third agent and then transfers the conference. The diagrams illustrate the call topology when the customer is present throughout the call (see page 83) and when the customer leaves the call before the second agent consults the third agent (see page 84).
- An agent consults to another agent and then conferences the call. Subsequently, the second agent consults to a third agent and then conferences the third agent. The diagrams illustrate the call topology when the customer is present throughout the call (see page 85) and when the customer leaves the call before the second agent consults the third agent (see page 86).

Inbound to Agent via ACD Queue

In this call topology, an inbound call is delivered to an agent via an ACD queue. The interaction arrives at the ACD queue, and the ACD queue diverts it to an agent.

Figure 4 depicts the call topology.

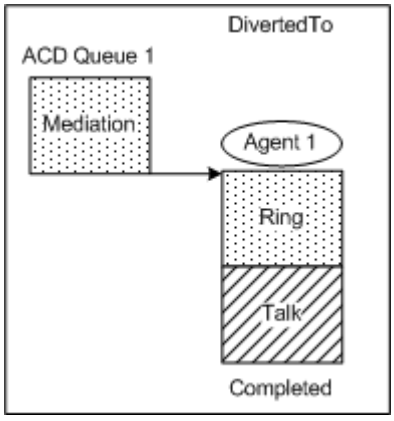

Figure 4: Inbound to Agent via ACD Queue

Inbound to Agent Directly

In this call topology, an inbound call is answered directly by an agent.

Figure 5 depicts the call topology.

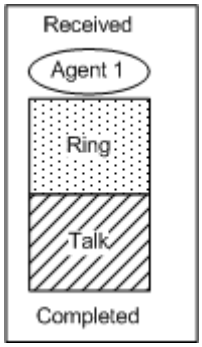

Figure 5: Inbound to Agent Directly

Mute Transfer to ACD Queue

In this call topology, an inbound call arrives at the ACD queue and is diverted to an agent. The agent then mute transfers the call to another ACD queue.

This section shows three possible outcomes of a call that is mute transferred to an ACD queue:

- The call is abandoned while it is in the second ACD queue (see "Mute Transfer to ACD Queue—Abandoned in Queue" on page 72).
- The call is abandoned while it is ringing at the second agent (see "Mute Transfer to ACD Queue—Abandoned While Ringing" on page 72).
- The call is successfully transferred to the second agent (see "Mute Transfer to ACD Queue—Completed" on page 73).

Mute Transfer to ACD Queue—Abandoned in Queue

For this outcome, the call is abandoned while it is in the second ACD queue.

Figure 6 depicts the call topology.

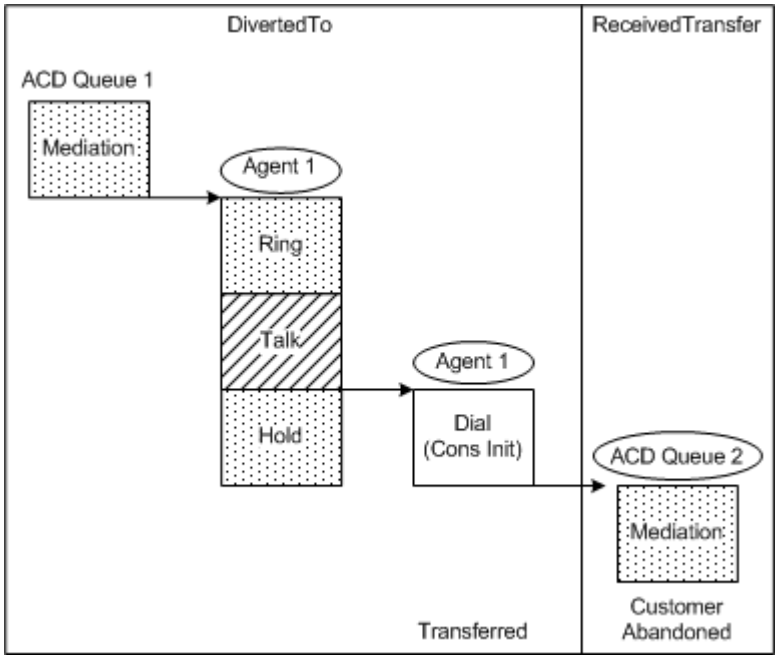

Figure 6: Transfer Abandoned While in ACD Queue

Mute Transfer to ACD Queue—Abandoned While Ringing

For this outcome, the call is diverted to the second agent, but it is abandoned while ringing.

Figure 7 on page 73 depicts the call topology.

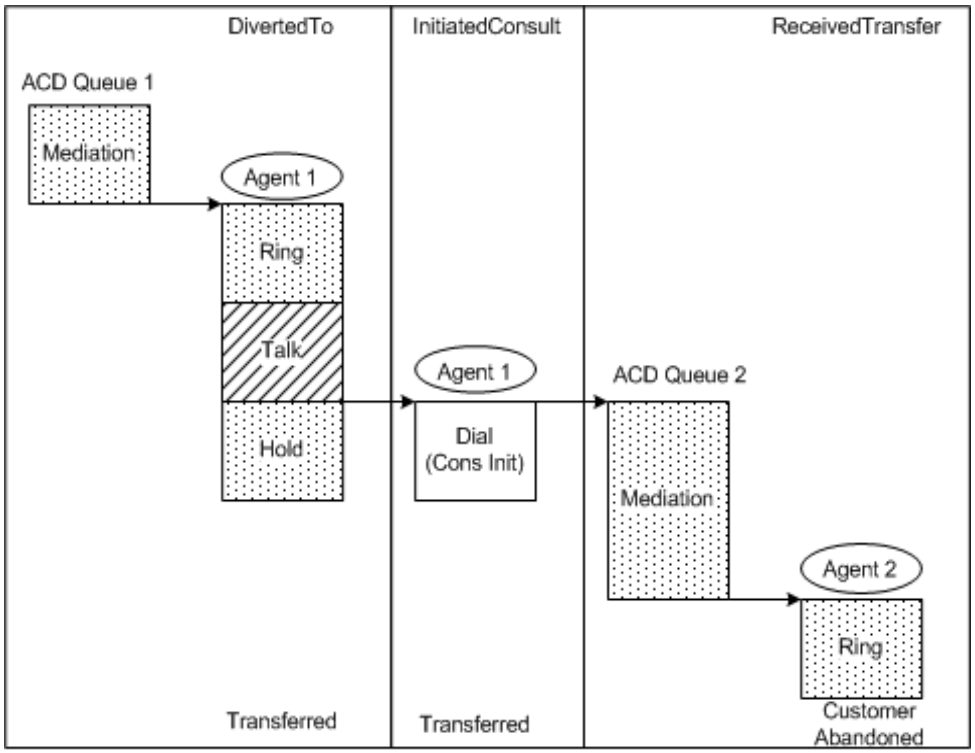

Figure 7: Transfer Abandoned While Ringing at Agent

Mute Transfer to ACD Queue—Completed

For this outcome, the call is successfully diverted to another agent.

Figure 8 on page 74 depicts the call topology.

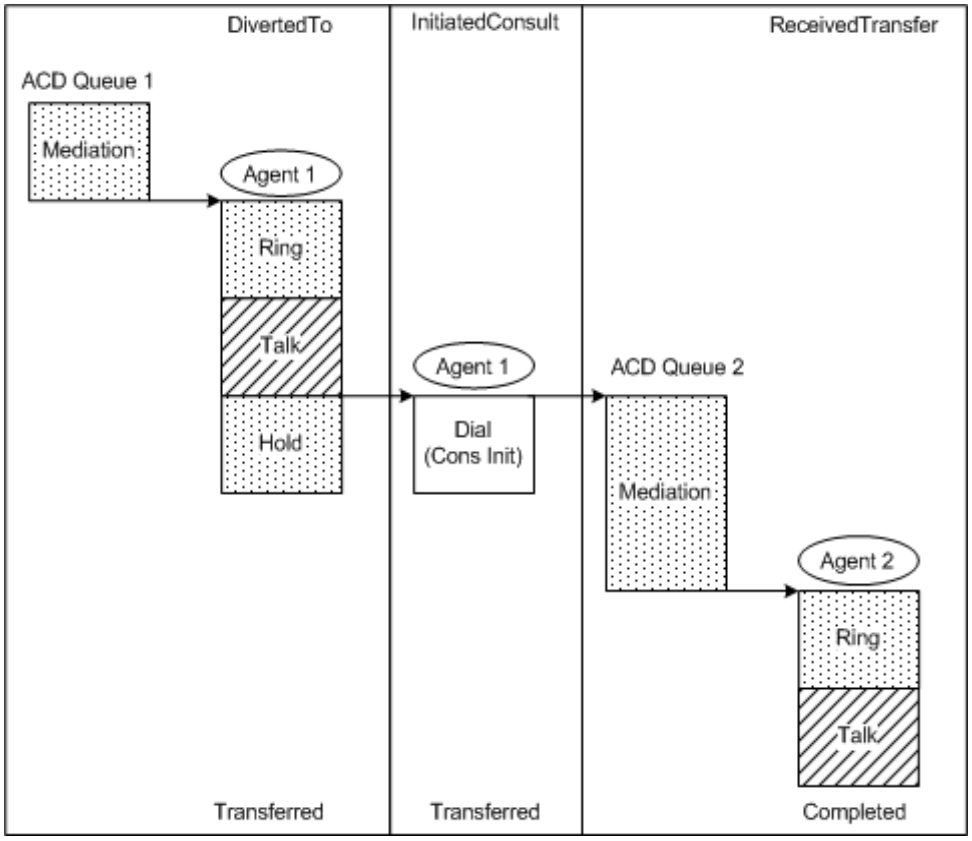

Figure 8: Transfer Completed

Mute Transfer to Agent

This call topology shows the outcome of a call that arrives at an agent, who answers the call and then mute transfers it to another agent.

Figure 9 on page 75 depicts the call topology.

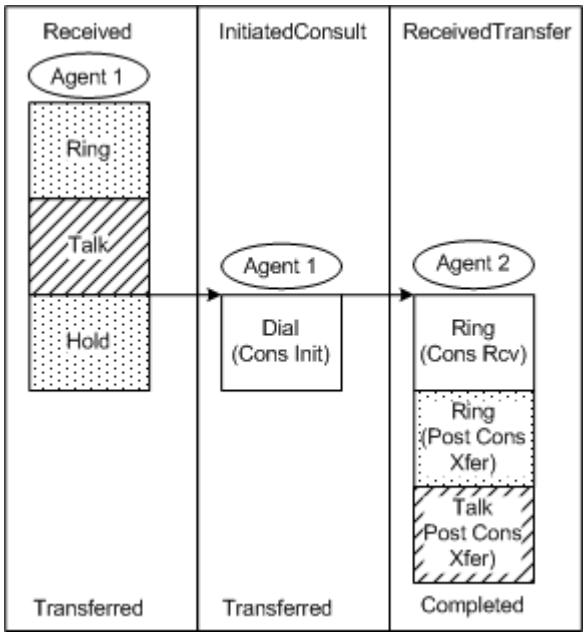

Figure 9: Agent Mute Transfers to Another Agent

Consult to Agent via ACD Queue, and Then Retrieve

In this call topology, an inbound call arrives at the ACD queue and is diverted to an agent. The agent consults to another ACD queue, and the call is diverted to another agent. The consultation ends when the first agent retrieves the call.

This section shows three possible outcomes of a call that is retrieved after a consultation has been initiated:

- The call is retrieved while it is in the second queue (see "Consult to ACD Queue—Abandoned in Queue").
- The call is retrieved while it is ringing at the second agent (see "Consult to ACD Queue—Abandoned While Ringing" on page 76).
- The call is retrieved after the consultation is completed (see "Consult to ACD Queue—Completed" on page 77).

Consult to ACD Queue—Abandoned in Queue

For this outcome, the call is retrieved while it is in the second ACD queue.

Figure 10 on page 76 depicts the call topology. Note that from the IRF perspective, the call is abandoned from the queue because no new handling resource receives the call from the queue.

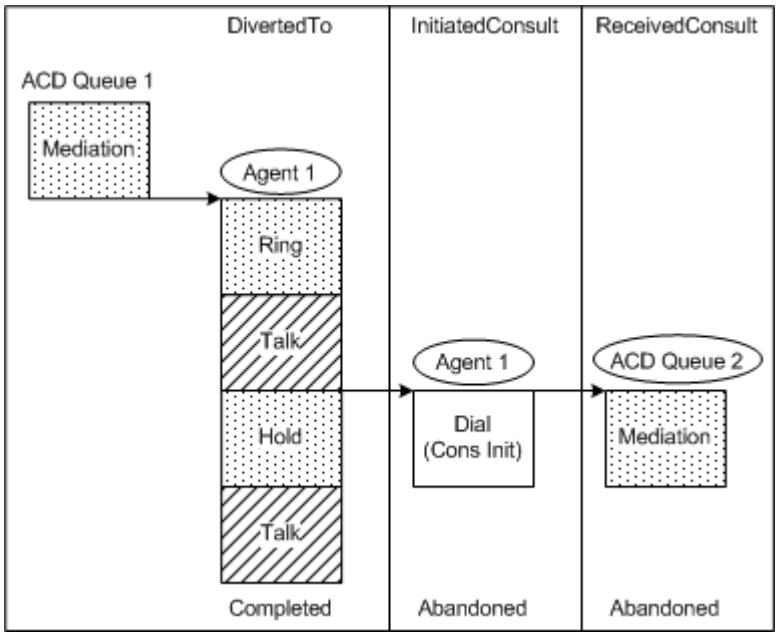

Figure 10: Call Retrieved While in ACD Queue

Consult to ACD Queue—Abandoned While Ringing

For this outcome, the call is retrieved while it is ringing at the second agent.

Figure 11 on page 77 depicts the call topology. Note that from the IRF perspective, the call is abandoned from the queue because the new handling resource, Agent 2, never receives the call.

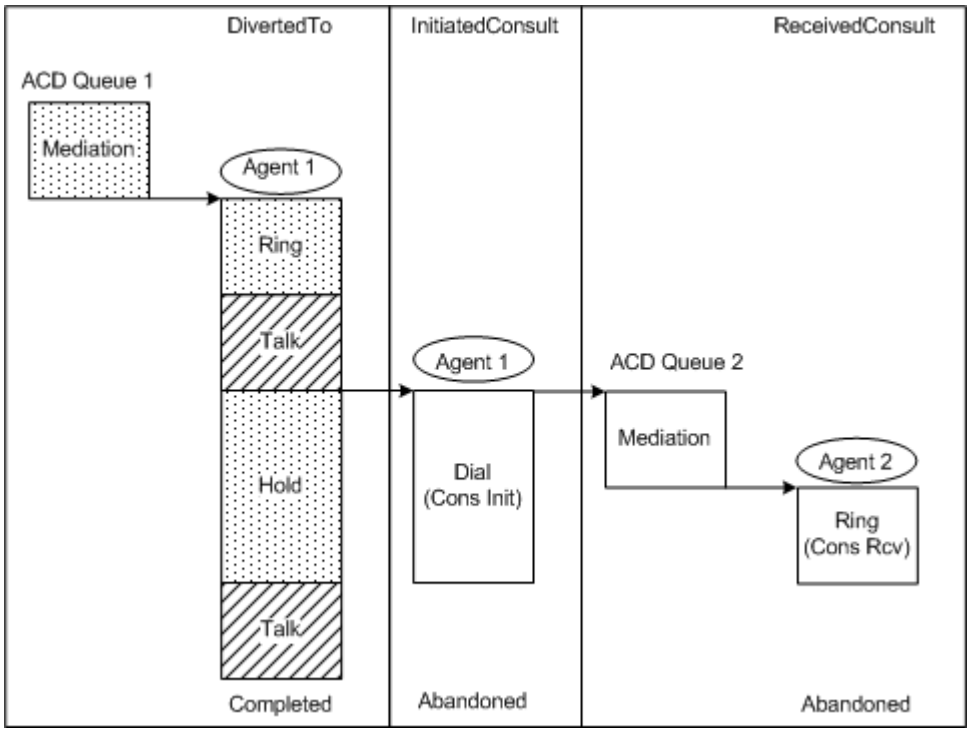

Figure 11: Call Retrieved While Ringing at Agent

Consult to ACD Queue—Completed

For this outcome, the call is retrieved after the consultation is completed. Figure 12 on page 78 depicts the call topology.

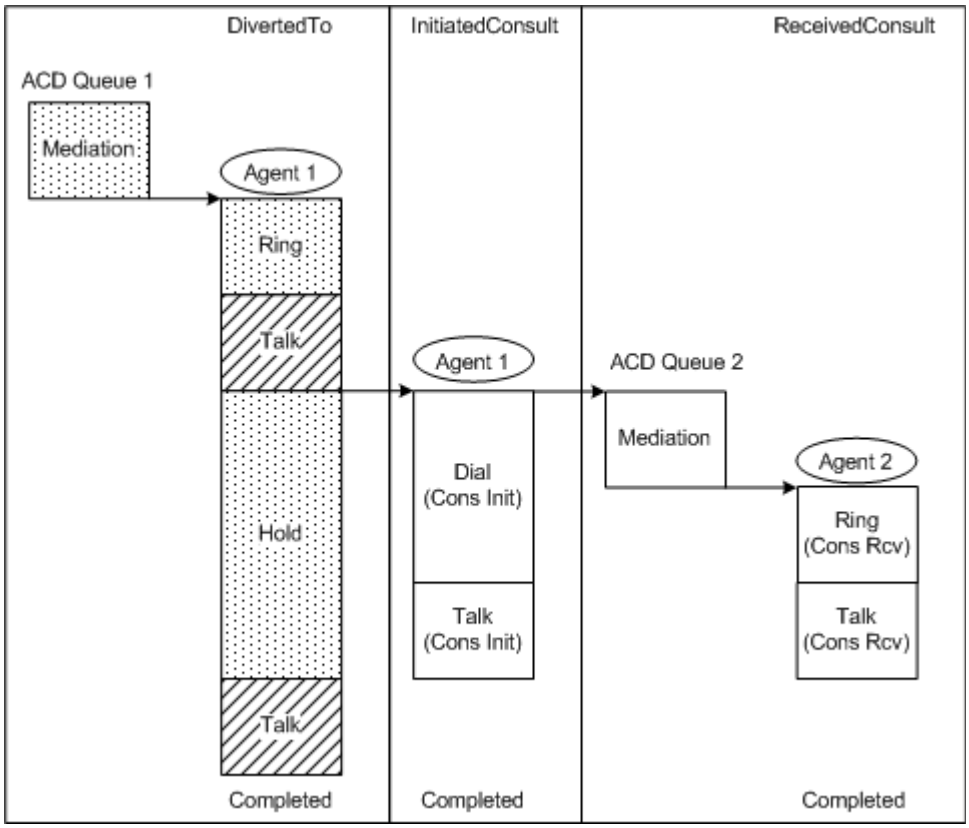

Figure 12: Consultation Completed

Consult to Agent, and Then Retrieve

This call topology shows the outcome of a call that arrives at an agent, who consults to another agent. The consultation ends when the first agent retrieves the call.

Figure 13 on page 79 depicts the call topology.

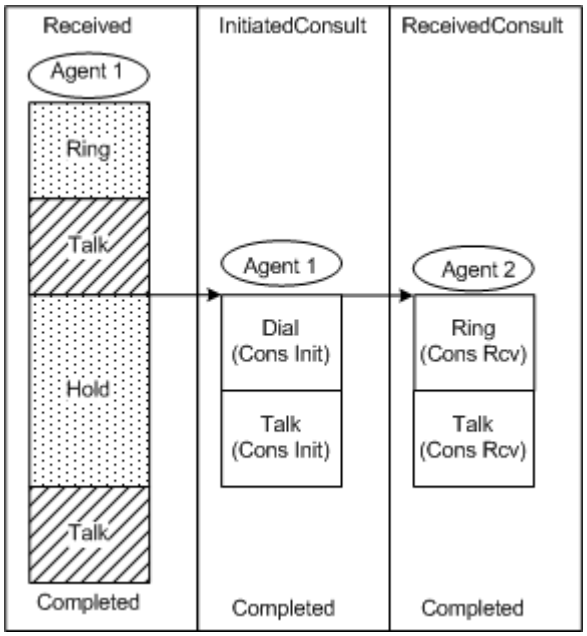

Figure 13: Consult to Agent, and Then Retrieve

Consult to Agent via ACD Queue, and Then Transfer

This call topology shows the outcome of a call that is transferred after a consultation. The call arrives at the ACD queue and is diverted to an agent. The agent consults to another ACD queue, and the call is diverted to another agent. The consultation ends when the first agent transfers the call.

Figure 14 on page 80 depicts the call topology.

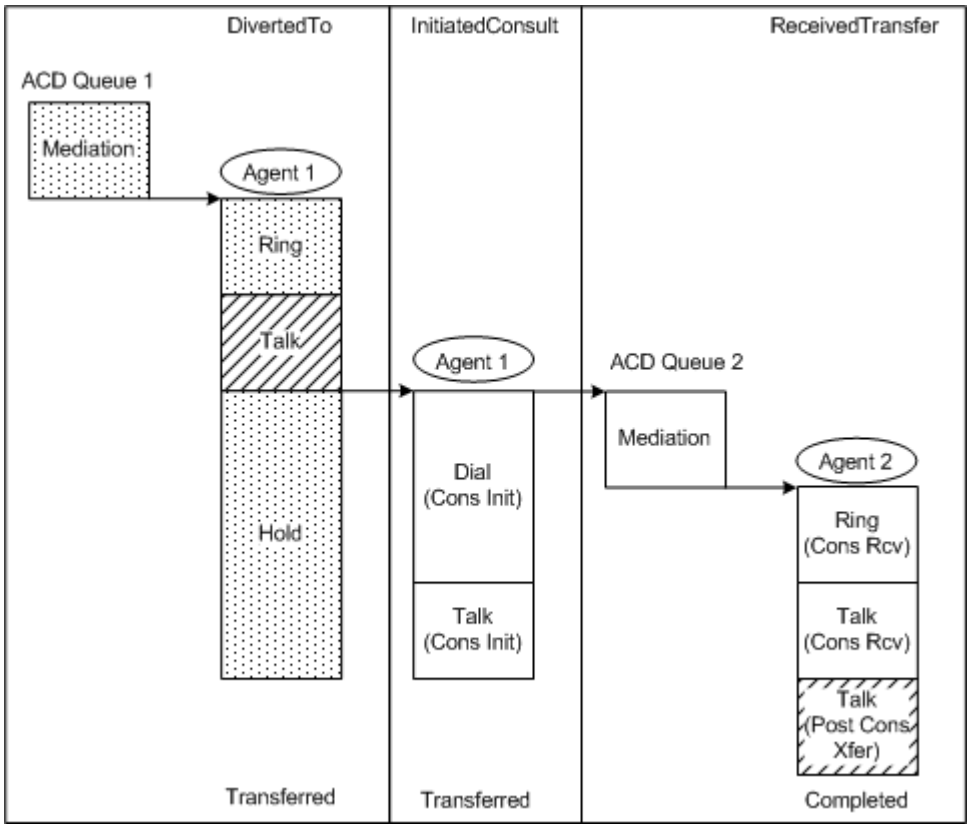

Figure 14: Consult to Agent via ACD Queue, and Then Transfer

Consult to Agent Directly, and Then Transfer

This call topology shows the outcome of a call that is transferred after a consultation. The call arrives at an agent, who consults to another agent, and then transfers the call. The consultation ends when the first agent transfers the call.

Figure 15 on page 81 depicts the IRF representation of the call topology.

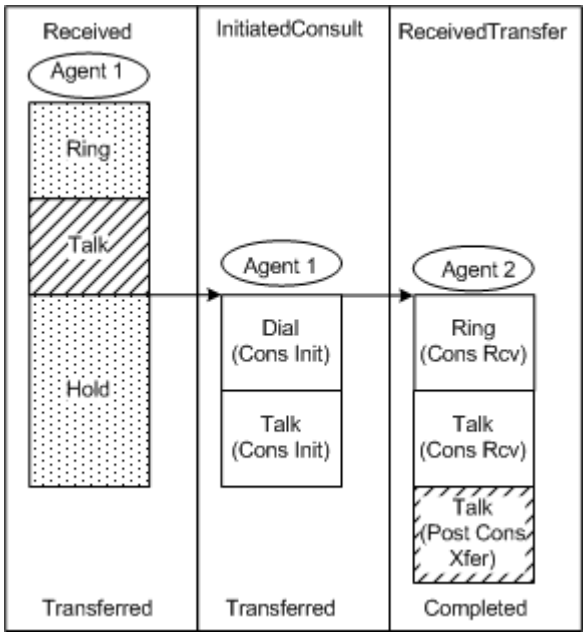

Figure 15: Consult to Agent Directly, and Then Transfer

Consult to Agent via ACD Queue, and Then Conference

This call topology shows the outcome of a call that is conferenced after a consultation. The call arrives at the ACD queue and is diverted to an agent. The agent consults to another ACD queue, and the call is diverted to another agent. The consultation ends when the first agent conferences the call.

Figure 16 on page 82 depicts the IRF representation of the call topology.

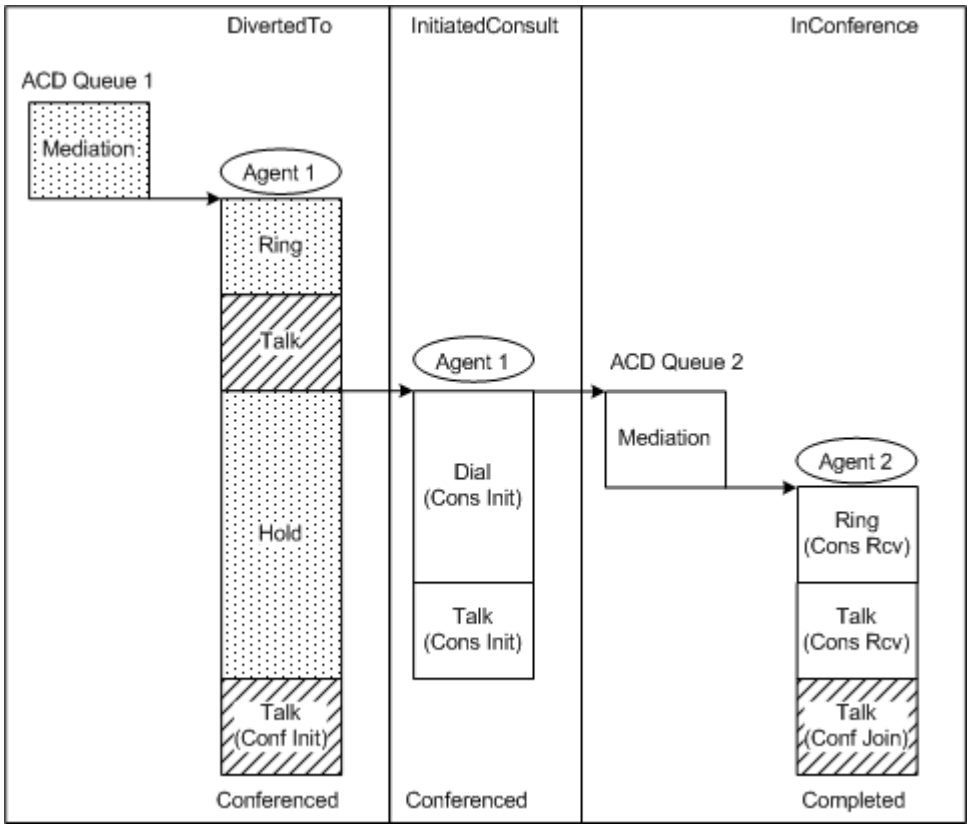

Figure 16: Consult to Agent via ACD Queue, and Then Conference

Consult to Agent Directly, and Then Conference

This call topology shows the outcome of a call that is conferenced after a consultation. The call arrives at an agent, who consults to another agent. The consultation ends when the first agent conferences the call.

Figure 17 on page 83 depicts the IRF representation of the call topology.

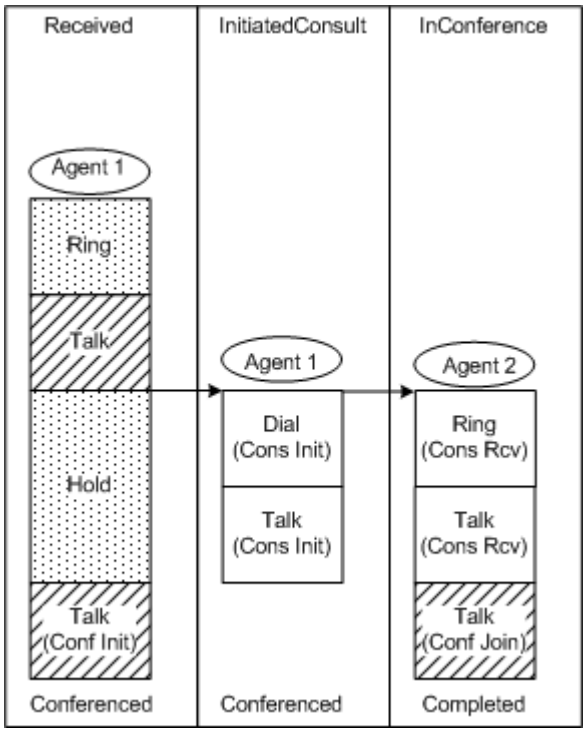

Figure 17: Consult to Agent Directly, and Then Conference

Consult and Transfer of a Conference—Customer Present Throughout

This call topology shows the outcome of the transfer of a conference. The call arrives at an agent, who consults to another agent and then conferences the call. The second agent then consults to a third agent. The consultation ends when the second agent transfers the conference to the third agent. The customer is present for the entire duration of the call.

Figure 18 on page 84 depicts the IRF representation of the call topology. Figure 19 on page 85 depicts the same call topology when the customer leaves the call before the second agent consults the third agent, to illustrate the effect on customer metrics.

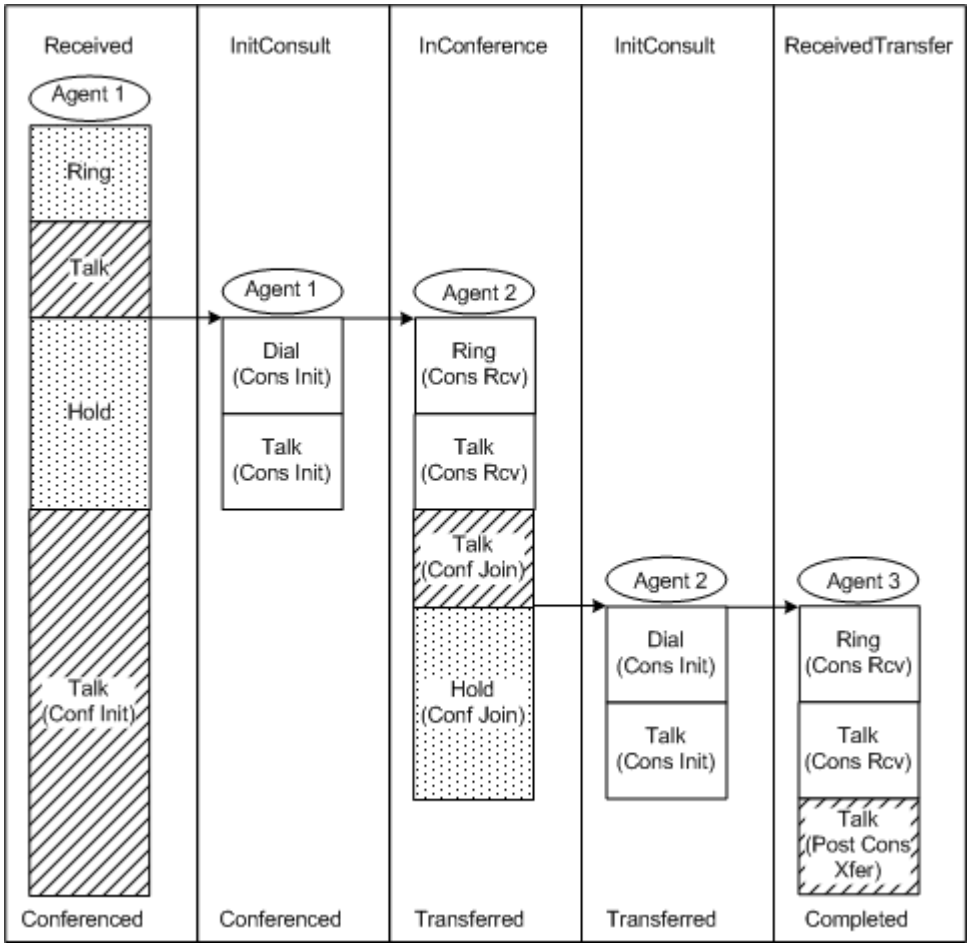

Figure 18: Consult and Transfer of a Conference (Customer Present Throughout)

Consult and Transfer of a Conference—Customer Leaves

This call topology shows the outcome of the transfer of a conference. The call arrives at an agent, who consults to another agent and then conferences the call. The second agent then consults to a third agent. The consultation ends when the second agent transfers the conference to the third agent. The customer leaves the call before the second agent consults the third agent.

Figure 19 depicts the IRF representation of the call topology. Figure 18 on page 84 depicts the same call topology when the customer is present for the entire duration of the call, to illustrate the effect on customer metrics.

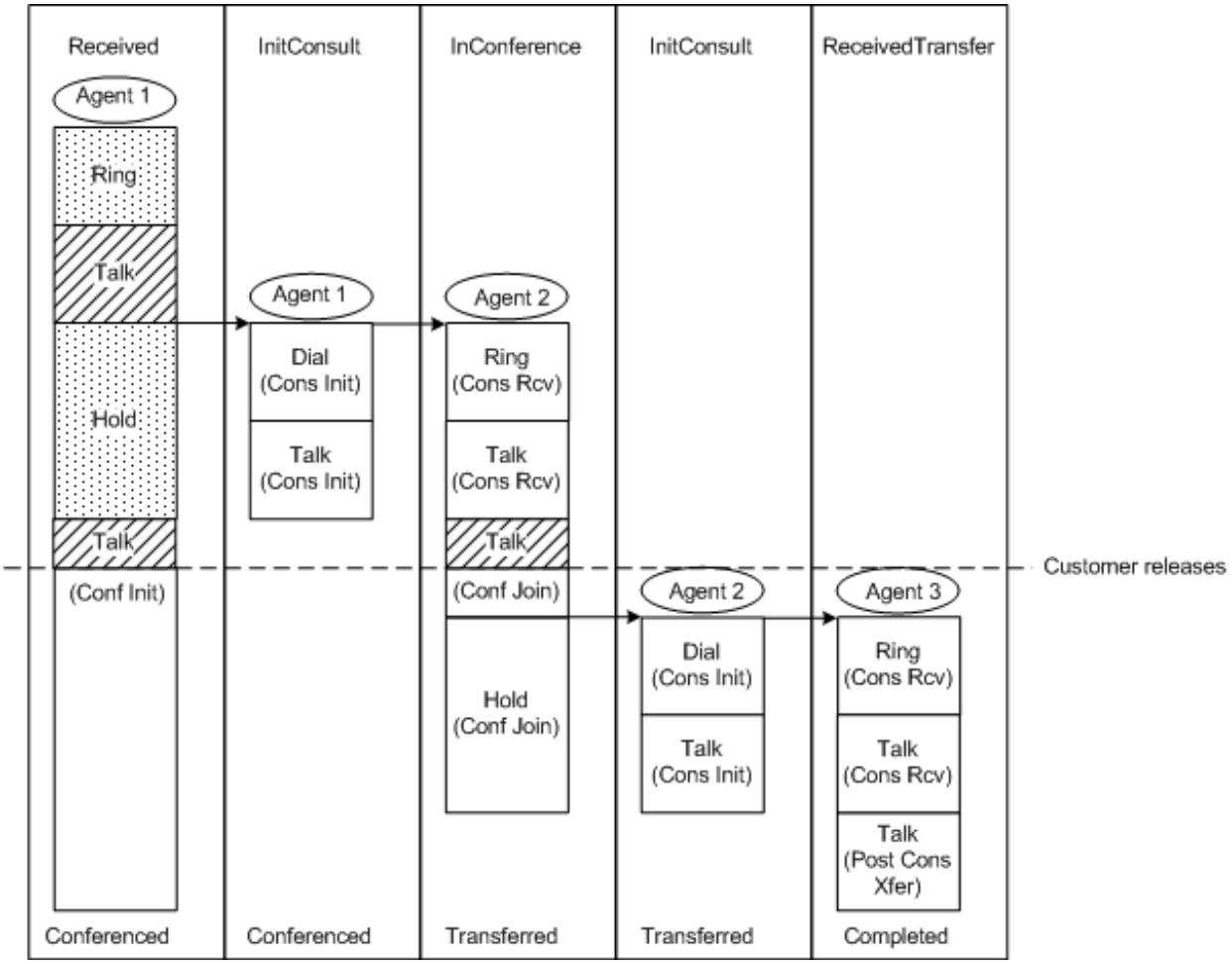

Figure 19: Consult and Transfer of a Conference (Customer Leaves During First Conference)

Consult and Conference of a Conference—Customer Present Throughout

This call topology shows the outcome of the conference of a conference. The call arrives at an agent, who consults to another agent and then conferences the call. The second agent then consults to a third agent. The consultation ends when the second agent conferences the third agent. The customer is present for the entire duration of the call.

Figure 20 on page 86 depicts the IRF representation of the call topology. Figure 21 on page 87 depicts the same call topology when the customer leaves

the call before the second agent consults the third agent, to illustrate the effect on customer metrics.

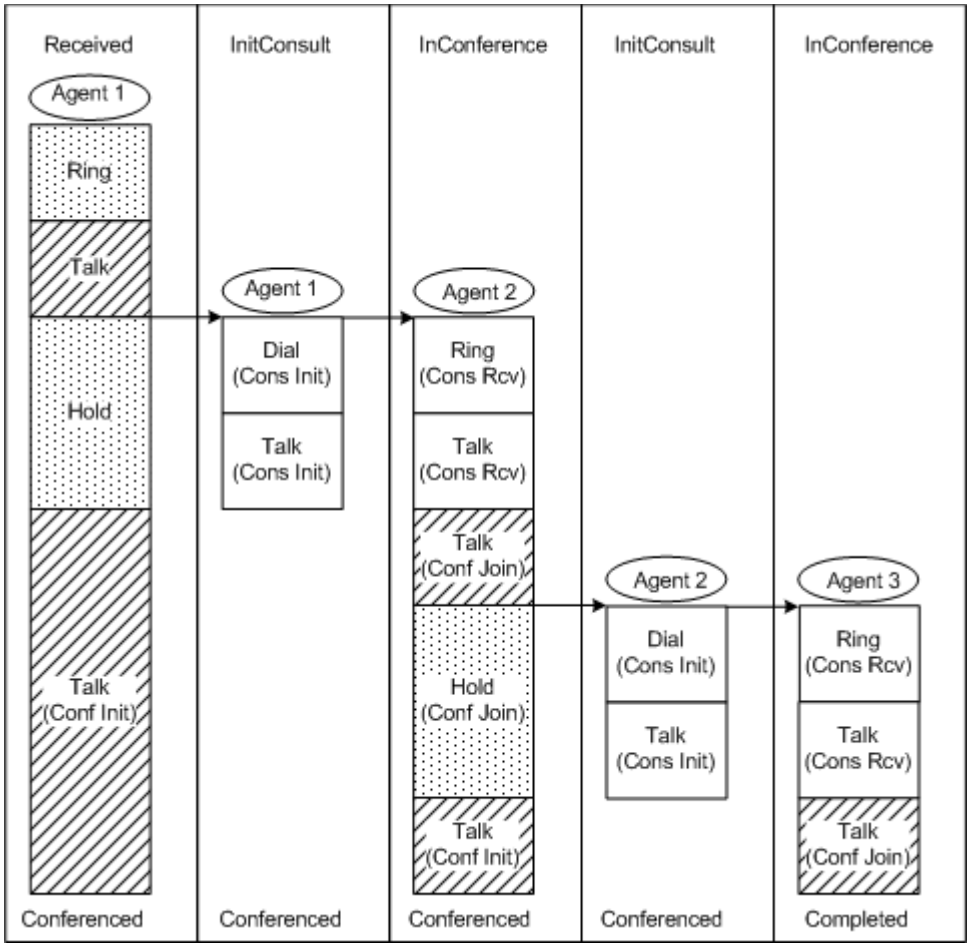

Figure 20: Consult and Conference of a Conference (Customer Present Throughout)

Consult and Conference of a Conference—Customer Leaves

This call topology shows the outcome of the conference of a conference. The call arrives at an agent, who consults to another agent and then conferences the call. The second agent then consults to a third agent. The consultation ends when the second agent conferences the third agent. The customer leaves the call before the second agent consults the third agent.

Figure 21 depicts the IRF representation of the call topology. Figure 20 on page 86 depicts the same call topology when the customer is present for the entire duration of the call, to illustrate the effect on customer metrics.

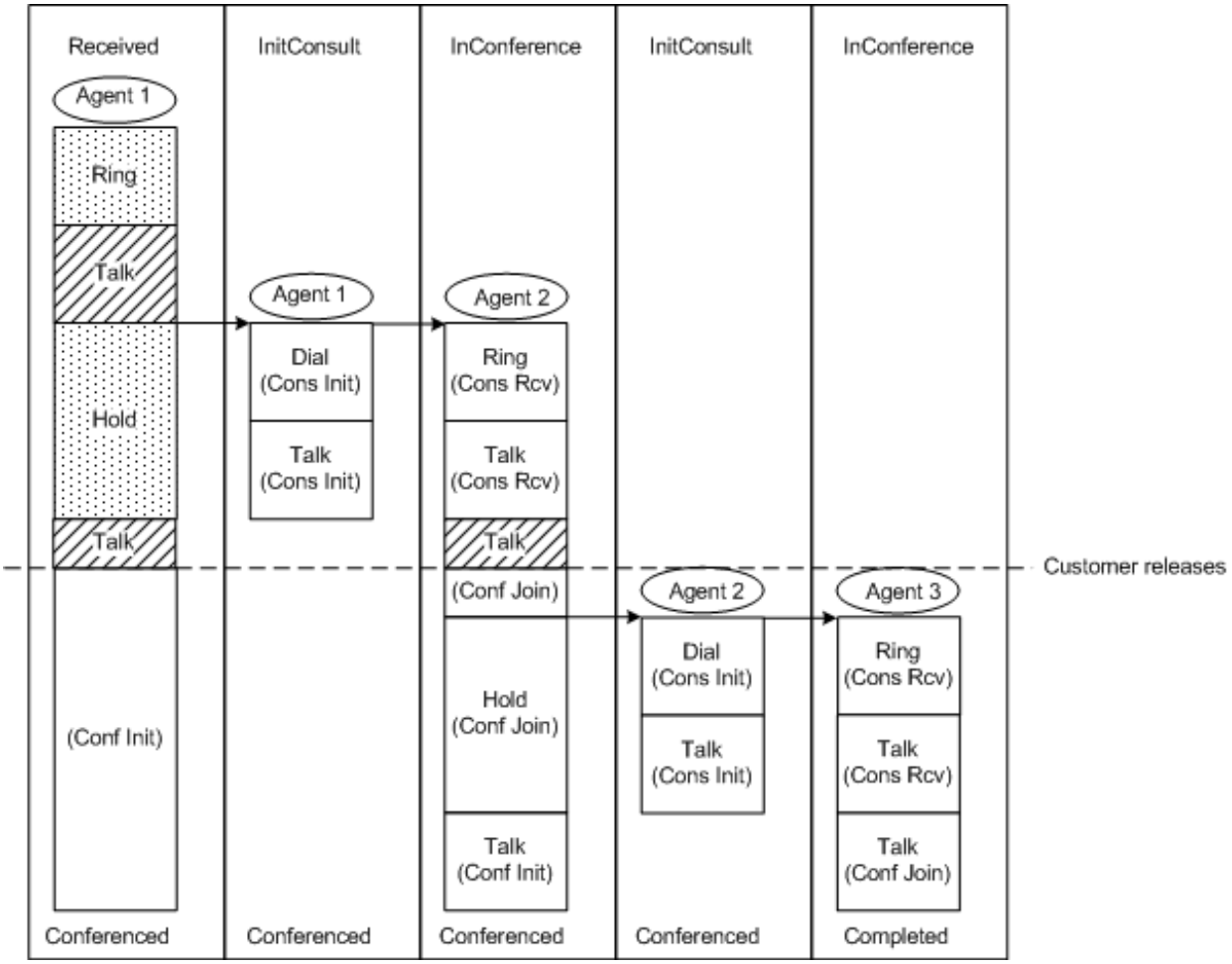

Figure 21: Consult and Conference of a Conference (Customer Leaves During First Conference)

Outbound Call Flow Example

This call topology shows a call flow example of a direct outbound call. An agent dials an off-switch number. After talking with an external party, the agent hangs up.

Figure 22 on page 88 depicts the IRF representation of the call topology.

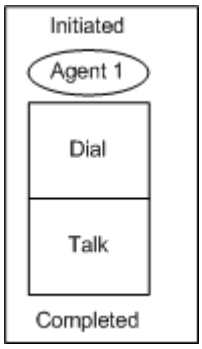

Figure 22: Agent Dials External Party

Internal Call Flow Examples

This subsection contains several examples of internal call flows. Each example represents a different outcome:

- An ACD queue directs the internal call to another agent.
- The internal call is answered directly by another agent (see page 89).
- An agent mute transfers the call to an ACD queue (see page 89).
- An agent mute transfers the call to another agent (see page 91).
- An agent consults to an ACD queue, and then retrieves the call (see page 93).
- An agent consults to another agent, and then retrieves the call (see page 94).
- An agent consults to an ACD queue, and then transfers the call (see page 96).
- An agent consults to another agent, and then transfers the call (see page 97).
- An agent consults to an ACD queue, and then conferences the call (see page 99).
- An agent consults to another agent, and then conferences the call (see page 100).

Internal to Agent via ACD Queue

This call topology shows the outcome of an internal call to an agent via an ACD queue. An agent initiates a call to the ACD queue, and the interaction is diverted to another agent.

Figure 23 on page 89 depicts the IRF representation of the call topology.

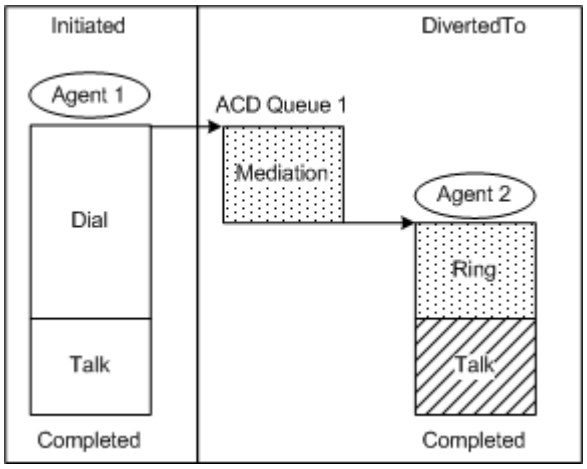

Figure 23: Internal Call to Another Agent via ACD Queue

Internal to Agent Directly

This call topology shows the outcome of a call that an agent initiates directly to another agent.

Figure 24 depicts the IRF representation of the call topology.

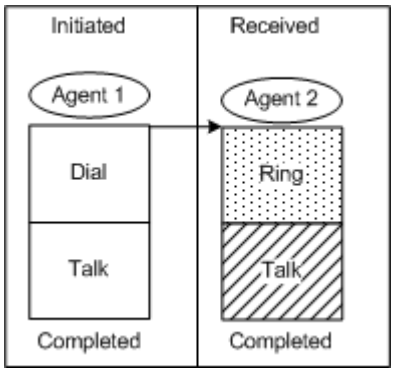

Figure 24: Internal Call from One Agent to Another

Mute Transfer to ACD Queue

In this call topology, an agent initiates a call to another agent. One of the agents then mute transfers the call to an ACD queue, and the interaction is diverted to another agent.

This section shows two possible outcomes of a call that is mute transferred to an ACD queue:

- The receiver (`Agent 2`) initiates the transfer (see “Mute Transfer to ACD Queue—Call Receiver Initiates Transfer” on page 90).
- The initiator (`Agent 1`) initiates the transfer (see “Mute Transfer to ACD Queue—Call Initiator Initiates Transfer” on page 90).

Mute Transfer to ACD Queue—Call Receiver Initiates Transfer

For this outcome, the receiving agent initiates a mute transfer to the ACD queue.

Figure 25 depicts the IRF representation of the call topology.

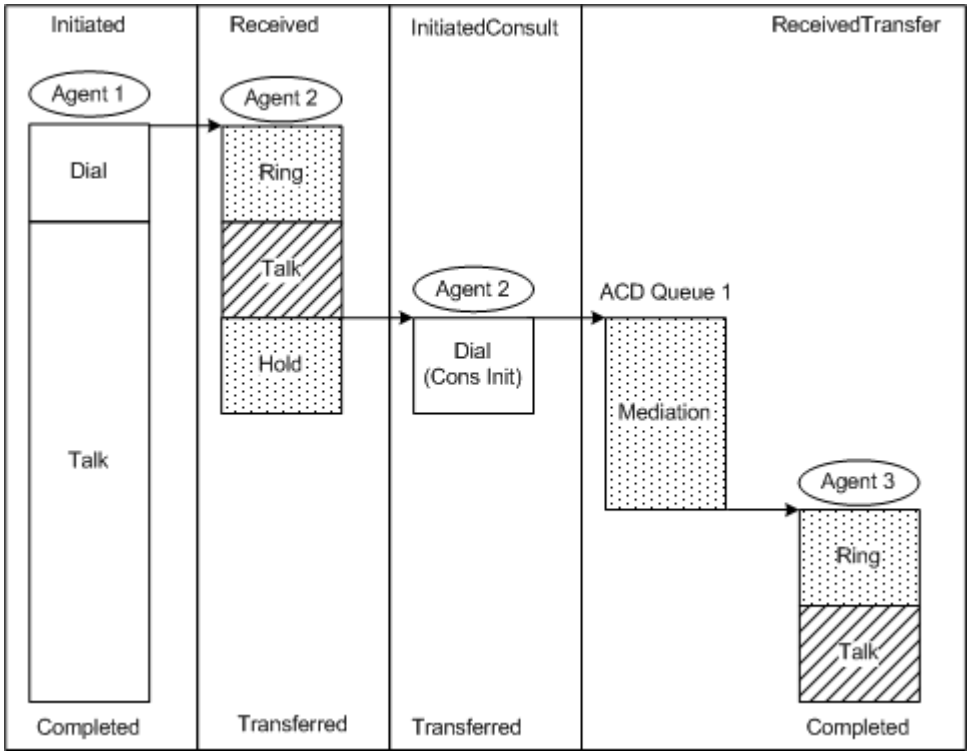

Figure 25: Receiving Agent Initiates Transfer to ACD Queue

Mute Transfer to ACD Queue—Call Initiator Initiates Transfer

For this outcome, the initiating agent initiates a mute transfer to the ACD queue.

Figure 26 on page 91 depicts the IRF representation of the call topology.

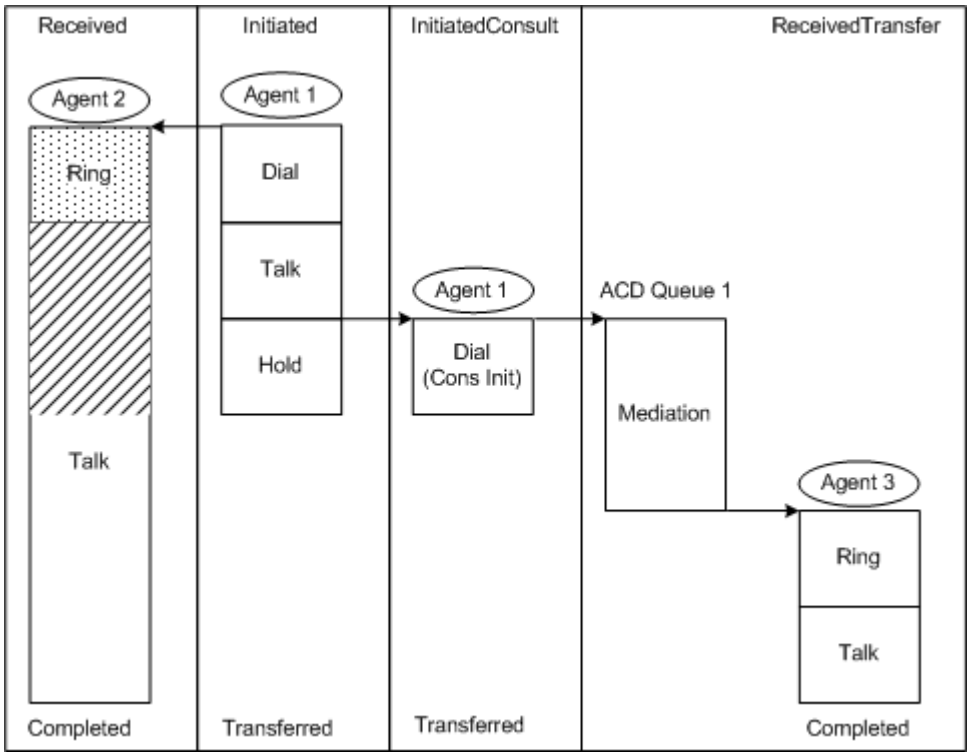

Figure 26: Initiating Agent Initiates Transfer to ACD Queue

Mute Transfer to Agent

In this call topology, an agent initiates a call to another agent. One of the agents then mute transfers the call to another agent.

This section shows two possible outcomes of a call that is mute transferred directly to an agent:

- The receiver (`Agent 2`) initiates the transfer (see “Mute Transfer to Agent—Call Receiver Initiates Transfer” on page 91).
- The initiator (`Agent 1`) initiates the transfer (see “Mute Transfer to Agent—Call Initiator Initiates Transfer” on page 92).

Mute Transfer to Agent—Call Receiver Initiates Transfer

For this outcome, the receiving agent initiates a mute transfer to another agent.

Figure 27 on page 92 depicts the IRF representation of the call topology.

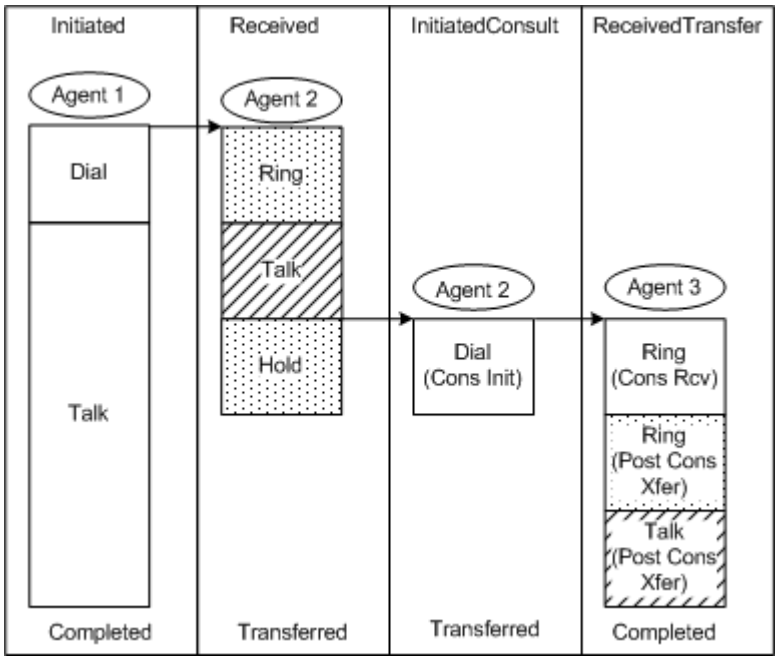

Figure 27: Receiving Agent Initiates Transfer to Another Agent

Mute Transfer to Agent—Call Initiator Initiates Transfer

For this outcome, the initiating agent initiates a mute transfer to another agent. Figure 28 depicts the IRF representation of the call topology.

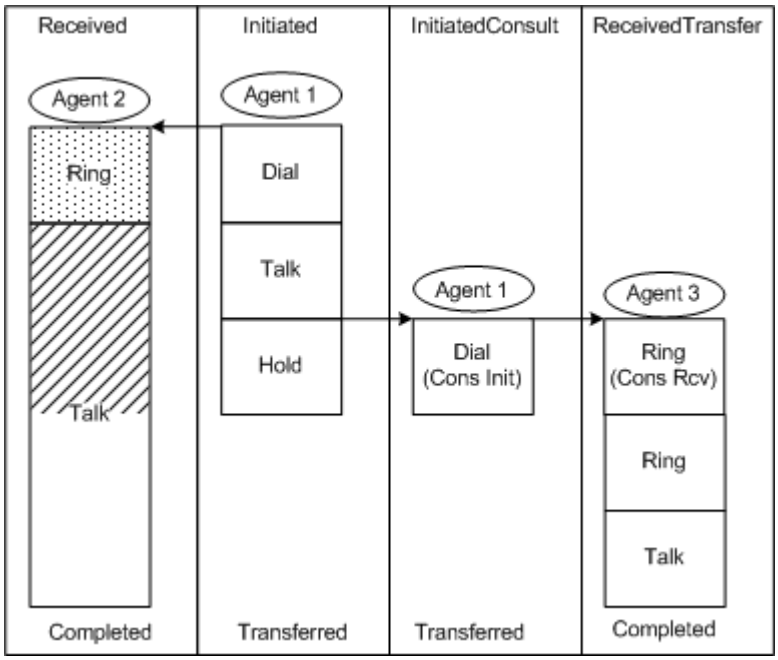

Figure 28: Initiating Agent Initiates Transfer to Another Agent

Consult to Agent via ACD Queue, and Then Retrieve

In this call topology, an agent initiates a call to another agent. One of the agents then initiates a consultation to an ACD queue, and the interaction is diverted to another agent. The consultation ends when the consulting agent retrieves the interaction.

This section shows two possible outcomes of a call that is retrieved after a consultation has been initiated:

- The receiver (`Agent 2`) initiates the consultation (see "Receiving Agent Consults to ACD Queue, and Then Retrieves" on page 93).
- The initiator (`Agent 1`) initiates the consultation (see "Initiating Agent Consults to ACD Queue, and Then Retrieves" on page 93)

Receiving Agent Consults to ACD Queue, and Then Retrieves

For this outcome, the receiving agent initiates a consultation to the ACD queue.

Figure 29 depicts the IRF representation of the call topology.

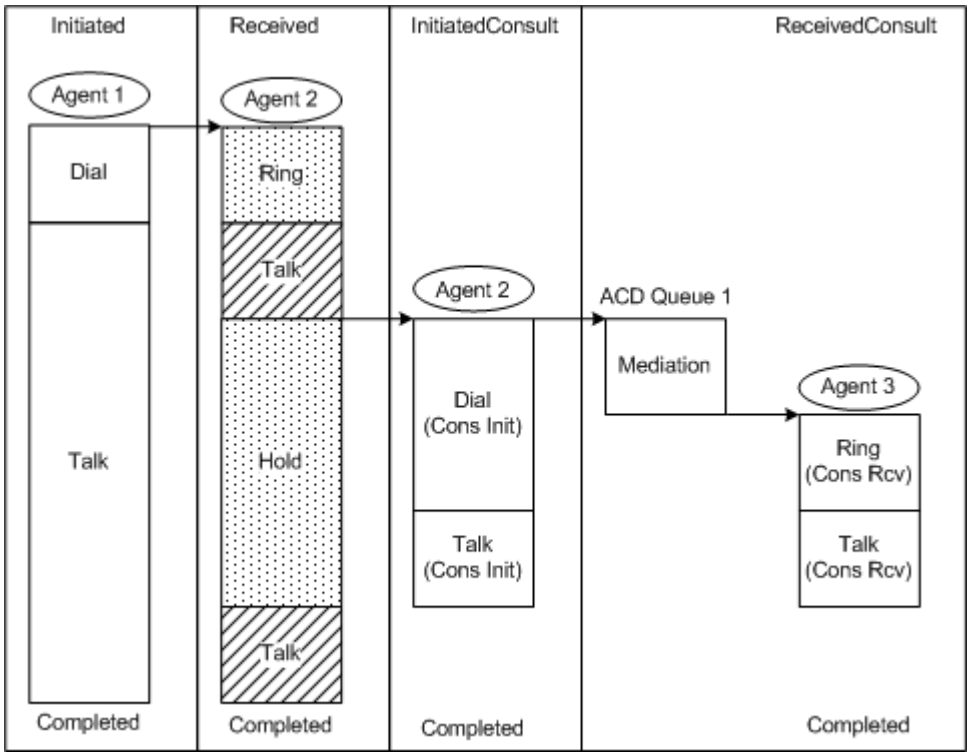

Figure 29: Receiving Agent Consults to ACD Queue Then Retrieves

Initiating Agent Consults to ACD Queue, and Then Retrieves

For this outcome, the initiating agent initiates a consultation to the ACD queue.

Figure 30 on page 94 depicts the IRF representation of the call topology.

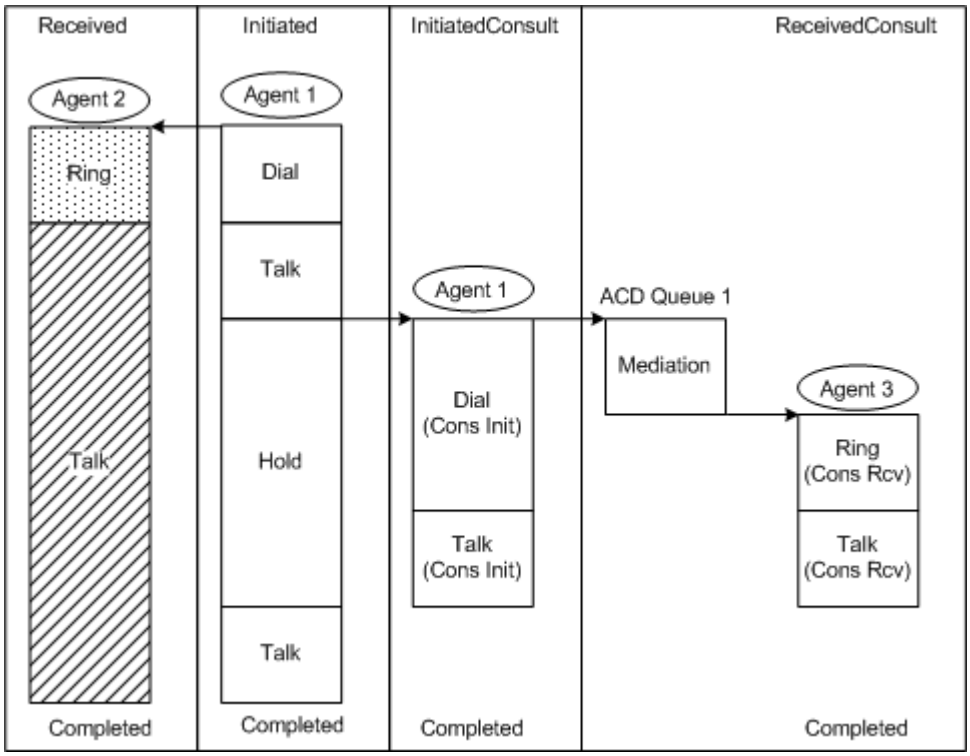

Figure 30: Initiating Agent Consults to ACD Queue Then Retrieves

Consult to Agent, and Then Retrieve

In this call topology, an agent initiates a call to another agent. One of the agents then initiates a consultation to a third agent. The consultation ends when the consulting agent retrieves the interaction.

This section shows two possible outcomes of a call that is retrieved after a consultation has been initiated:

- The receiver (`Agent 2`) initiates the consultation (see "Receiving Agent Consults to Another Agent, and Then Retrieves" on page 94).
- The initiator (`Agent 1`) initiates the consultation (see "Initiating Agent Consults to Another Agent, and Then Retrieves" on page 95)

Receiving Agent Consults to Another Agent, and Then Retrieves

For this outcome, the receiving agent initiates a consultation to another agent.

Figure 31 on page 95 depicts the IRF representation of the call topology.

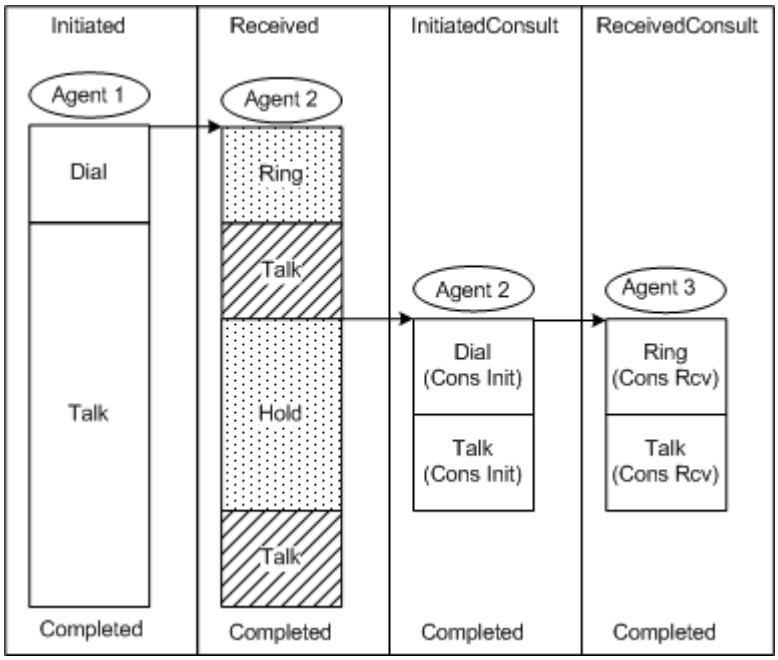

Figure 31: Receiving Agent Consults to Another Agent Then Retrieves

Initiating Agent Consults to Another Agent, and Then Retrieves

For this outcome, the initiating agent initiates a consultation to another agent. Figure 32 depicts the IRF representation of the call topology.

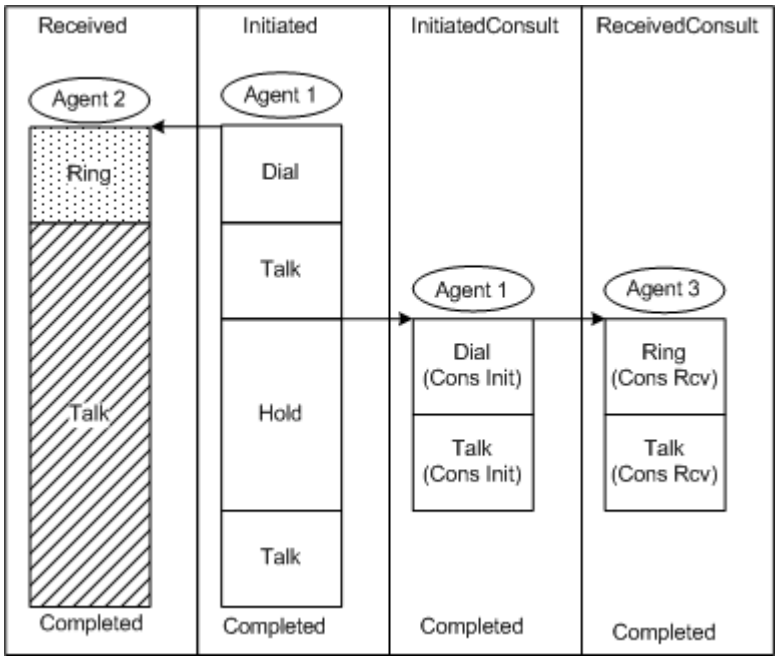

Figure 32: Initiating Agent Consults to Another Agent Then Retrieves

Consult to Agent via ACD Queue, and Then Transfer

In this call topology, an agent initiates a call to another agent. One of the agents then initiates a consultation to an ACD queue, and the interaction is diverted to another agent. The consultation ends when the consulting agent transfers the interaction.

This section shows two possible outcomes of a call that is transferred after a consultation:

- The receiver (`Agent 2`) initiates the consultation (see "Receiving Agent Consults to ACD Queue, and Then Transfers" on page 96).
- The initiator (`Agent 1`) initiates the consultation (see "Initiating Agent Consults to ACD Queue, and Then Transfers" on page 96)

Receiving Agent Consults to ACD Queue, and Then Transfers

For this outcome, the receiving agent initiates a consultation to the ACD queue.

Figure 33 depicts the IRF representation of the call topology.

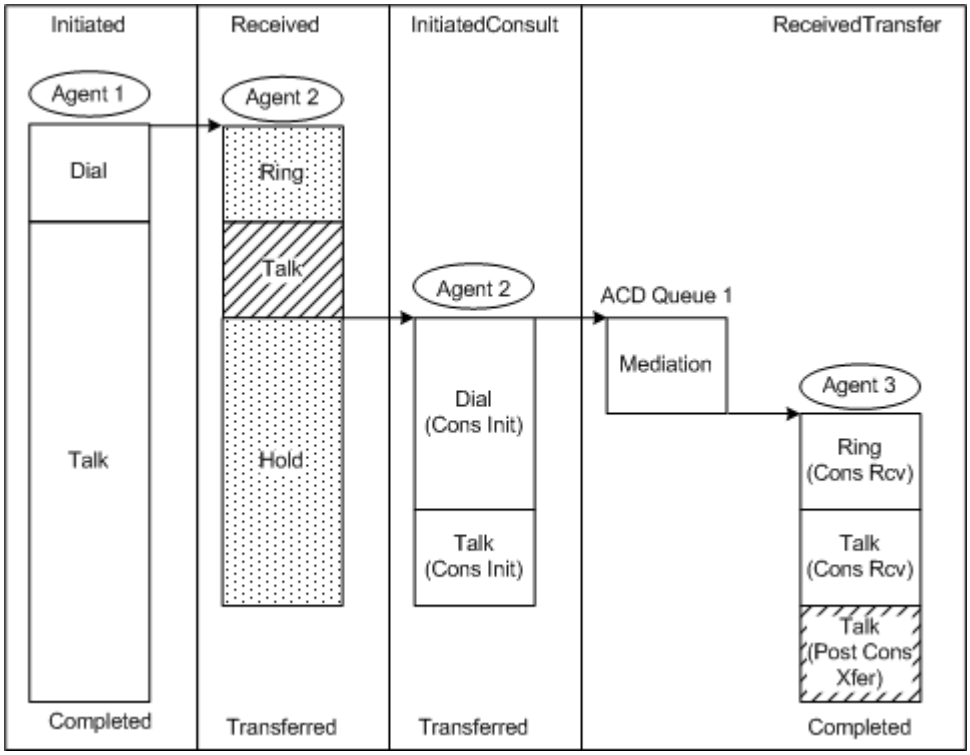

Figure 33: Receiving Agent Consults to ACD Queue Then Transfers

Initiating Agent Consults to ACD Queue, and Then Transfers

For this outcome, the initiating agent initiates a consultation to the ACD queue.

Figure 34 on page 97 depicts the IRF representation of the call topology.

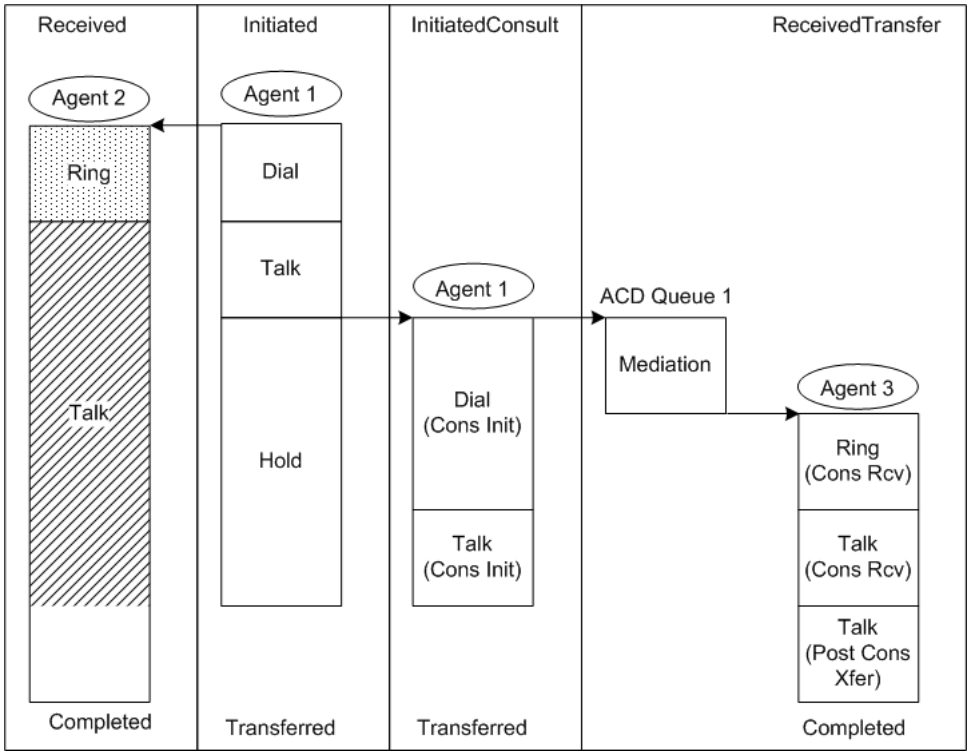

Figure 34: Initiating Agent Consults to ACD Queue Then Transfers

Consult to Agent, and Then Transfer

In this call topology, an agent initiates a call to another agent. One of the agents then initiates a consultation to a third agent. The consultation ends when the consulting agent transfers the interaction.

This section shows two possible outcomes of a call that is transferred after a consultation has been initiated:

- The receiver (`Agent 2`) initiates the consultation (see “Receiving Agent Consults to Another Agent, and Then Transfers” on page 97).
- The initiator (`Agent 1`) initiates the consultation (see “Initiating Agent Consults to Another Agent, and Then Transfers” on page 98)

Receiving Agent Consults to Another Agent, and Then Transfers

For this outcome, the receiving agent initiates a consultation to another agent.

Figure 35 on page 98 depicts the IRF representation of the call topology.

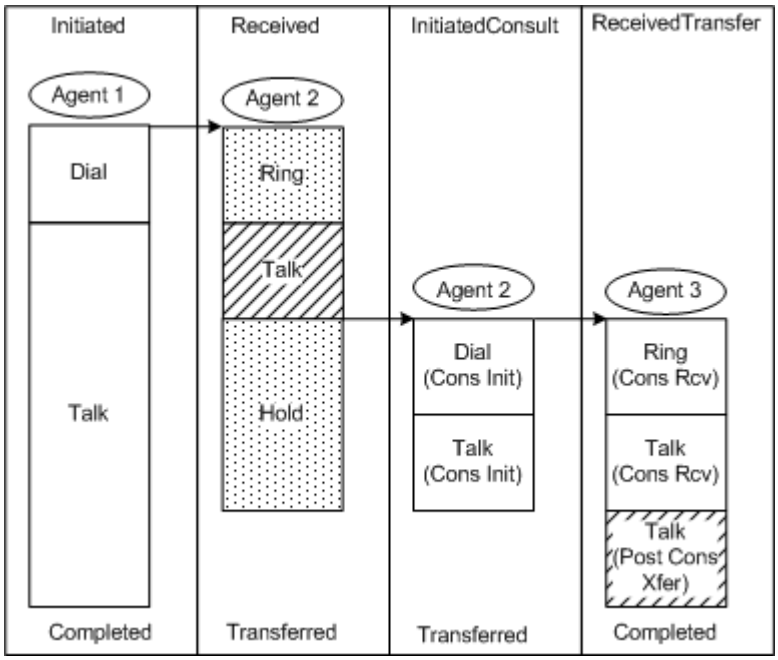

Figure 35: Receiving Agent Consults to Another Agent Then Transfers

Initiating Agent Consults to Another Agent, and Then Transfers

For this outcome, the initiating agent initiates a consultation to another agent. Figure 36 depicts the IRF representation of the call topology.

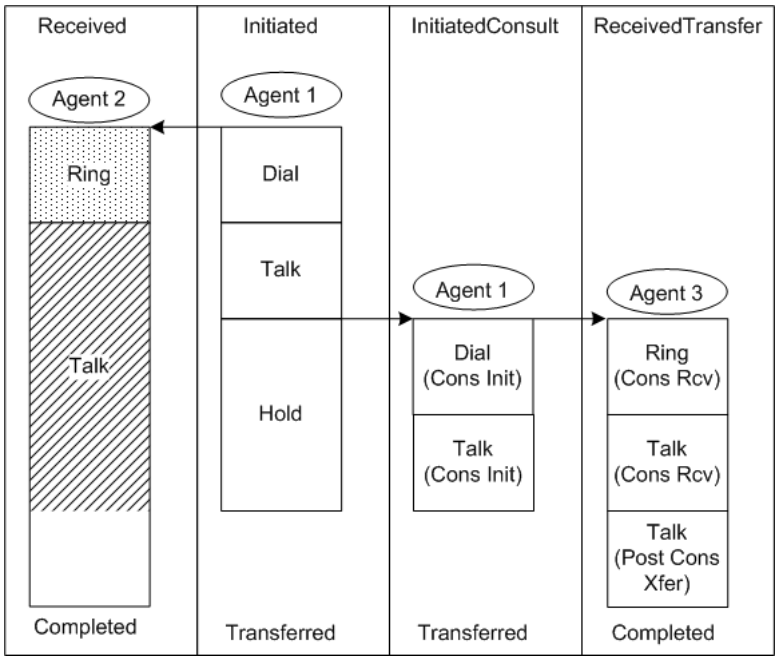

Figure 36: Initiating Agent Consults to Another Agent Then Transfers

Consult to Agent via ACD Queue, and Then Conference

In this call topology, an agent initiates a call to another agent. One agent then initiates a consultation to an ACD queue, and the interaction is diverted to a third agent. The consultation ends when the consulting agent conferences the interaction.

This section shows two possible outcomes of a call that is conferenced after a consultation:

- The receiver (`Agent 2`) initiates the consultation (see "Receiving Agent Consults to ACD Queue, and Then Conferences" on page 99).
- The initiator (`Agent 1`) initiates the consultation (see "Initiating Agent Consults to ACD Queue, and Then Conferences" on page 100).

Receiving Agent Consults to ACD Queue, and Then Conferences

For this outcome, the receiving agent initiates a consultation to the ACD queue.

Figure 37 depicts the IRF representation of the call topology.

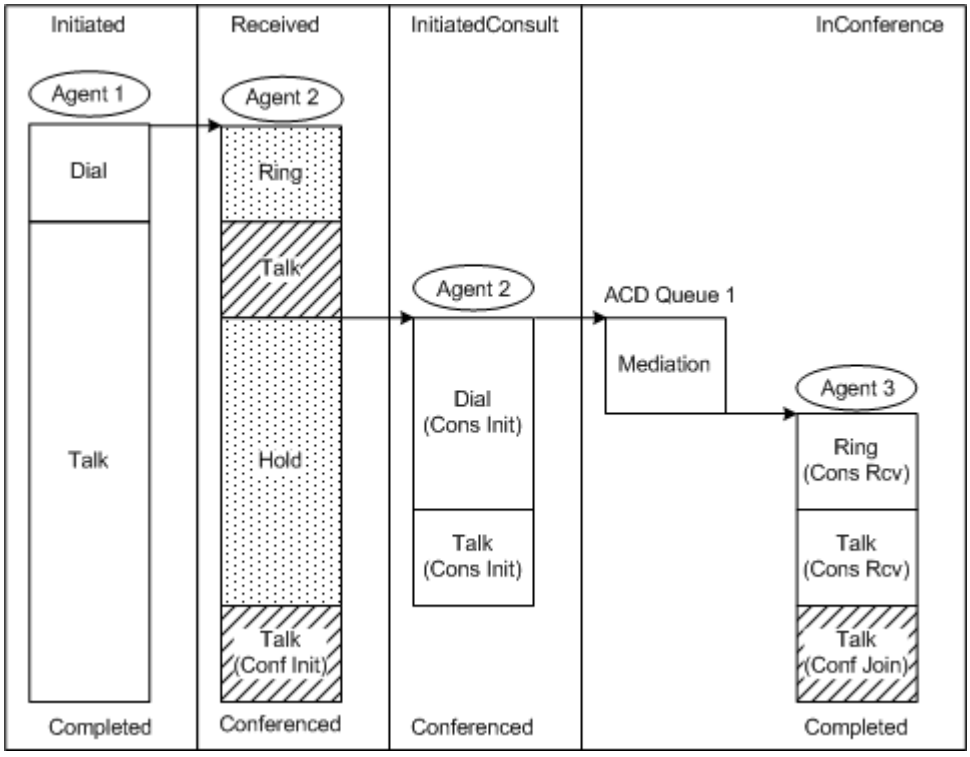

Figure 37: Receiving Agent Consults to ACD Queue Then Conferences

Initiating Agent Consults to ACD Queue, and Then Conferences

For this outcome, the initiating agent initiates a consultation to the ACD queue.

Figure 38 depicts the IRF representation of the call topology.

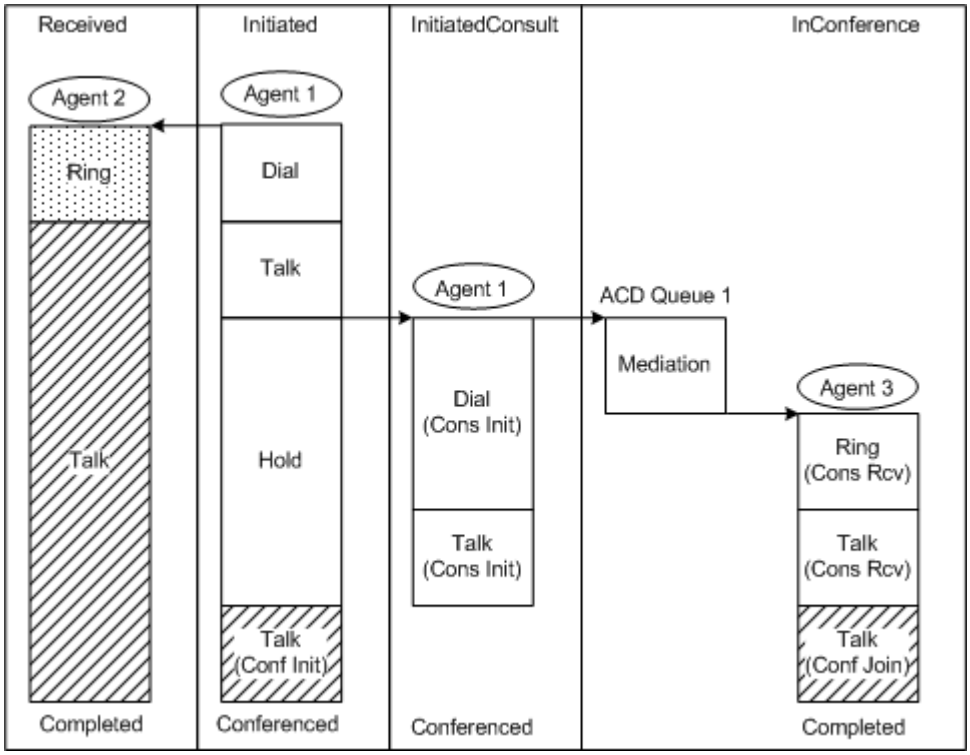

Figure 38: Initiating Agent Consults to ACD Queue Then Conferences

Consult to Agent, and Then Conference

In this call topology, an agent initiates a call to another agent. One agent then initiates a consultation to a third agent. The consultation ends when the consulting agent conferences the interaction.

This section shows two possible outcomes of a call that is conferenced after a consultation:

- The receiver (`Agent 2`) initiates the consultation (see "Receiving Agent Consults to Another Agent, and Then Conferences" on page 100).
- The initiator (`Agent 1`) initiates the consultation (see "Initiating Agent Consults to Another Agent, and Then Conferences" on page 101)

Receiving Agent Consults to Another Agent, and Then Conferences

For this outcome, the receiving agent initiates a consultation to another agent.

Figure 39 on page 101 depicts the IRF representation of the call topology.

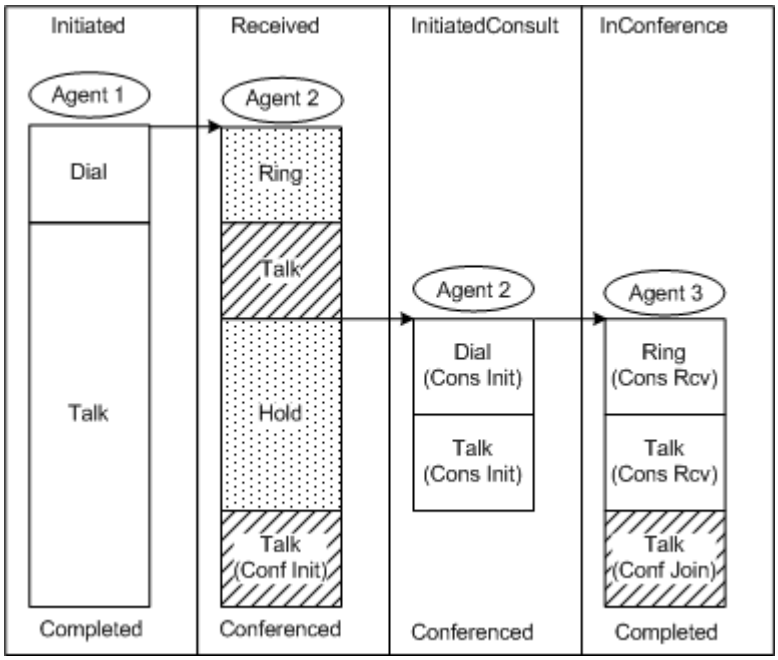

Figure 39: Receiving Agent Consults to Another Agent Then Conferences

Initiating Agent Consults to Another Agent, and Then Conferences

For this outcome, the initiating agent initiates a consultation to another agent. Figure 40 depicts the IRF representation of the call topology.

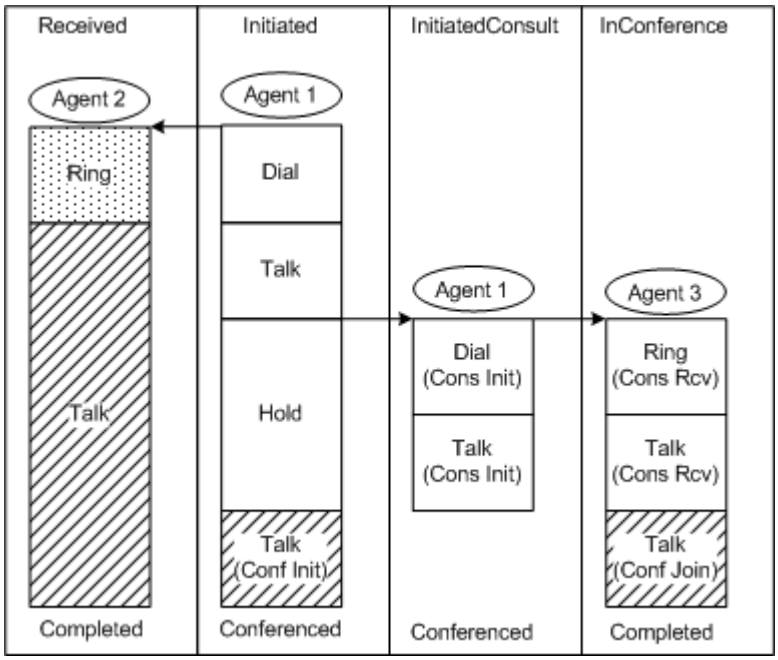

Figure 40: Initiating Agent Consults to Another Agent Then Conferences

IVR-in-Front-of-Switch Call Flows

Voice interactions arrive at an IVR that is visible to the IVR Server's virtual T-Server. Either self-service, or simply front-end identification and segmentation, can be the focus of the IVR application. If the IVR application cannot completely handle the voice interaction, the interaction can be transferred to an ACD queue behind the switch that represents a requested skill, service type, or customer segment. Agents who are logged in to the ACD queues handle the interactions.

Inbound Call Flow Examples

This subsection contains several examples of inbound call flows. Each example represents a different outcome:

- An inbound call arrives at an IVR DN (see page 102).
- The IVR transfers the call to an ACD queue (see page 103).
- The IVR transfers the call to an agent (see page 104).

Inbound to IVR DN

This call topology shows the outcomes of a call that arrives at an IVR DN.

- Figure 41 depicts the IRF representation of the call topology when the call completes normally in the case of a *self-service* (SS) IVR (when the IVR is in its own box).
- Figure 42 depicts the IRF representation of the call topology when the call is abandoned by the customer. This is the *nonself-service* (nonSS) IVR case.

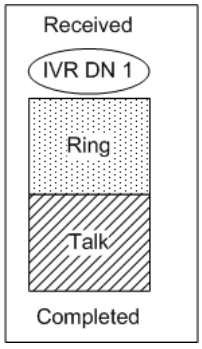

Figure 41: Inbound Interaction to IVR DN, Completed

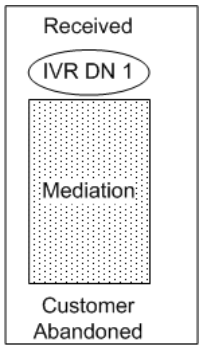

Figure 42: Inbound Interaction to IVR DN, Abandoned

IVR Transfer to ACD Queue

This call topology shows the outcome of an interaction that arrives at an IVR DN, which hook-flash transfers the interaction to an ACD queue.

- Figure 43 on page 103 depicts the IRF representation of the call topology in the case of an SS IVR (when the IVR is in its own box).
- Figure 44 on page 104 depicts the IRF representation of the call topology in the case of a nonSS IVR.

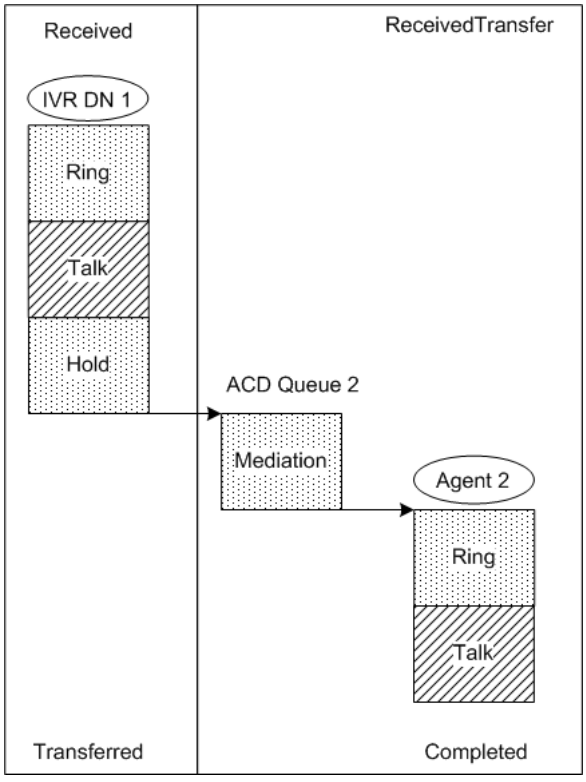

Figure 43: IVR Hook-Flash Transfer to ACD Queue (SS IVR)

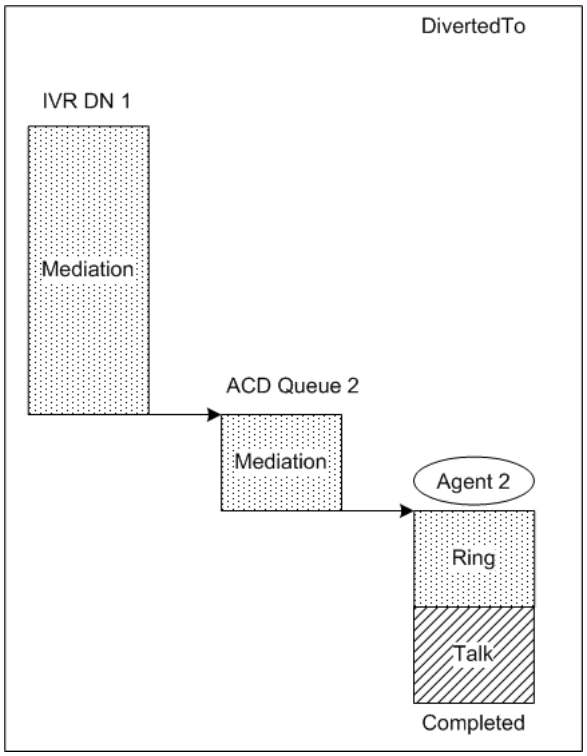

Figure 44: IVR Hook-Flash Transfer to ACD Queue (nonSS IVR)

IVR Transfer to Agent

This call topology shows the outcome of an interaction that arrives at an IVR DN, which hook-flash transfers the interaction to an agent.

- Figure 45 depicts the IRF representation of the call topology in the case of an SS IVR.
- Figure 46 depicts the IRF representation of the call topology in the case of a nonSS IVR.

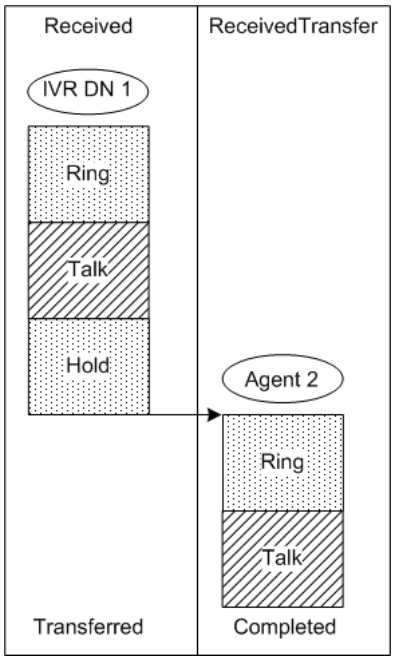

Figure 45: IVR Hook-Flash Transfer to Agent (SS IVR)

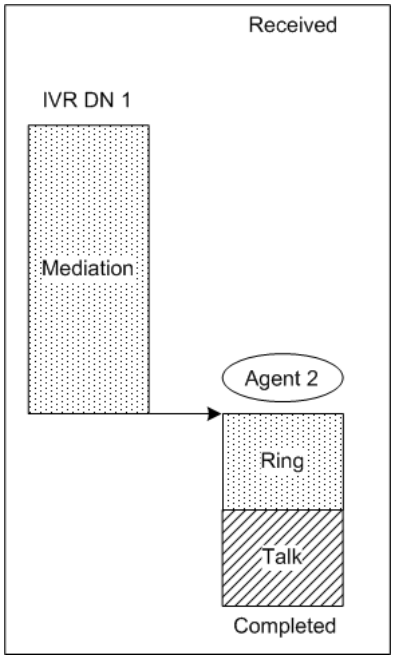

Figure 46: IVR Hook-Flash Transfer to Agent (nonSS IVR)

IVR-Behind-Switch Call Flows

Voice interactions that arrive at the switch are queued to an ACD queue, where the ACD positions are actually IVR DNs. Either self-service, or simply front-end identification and segmentation, can be the focus of the IVR application. If the IVR application cannot completely handle the voice interaction, the interaction can be transferred to an ACD queue that represents a requested skill, service type, or customer segment. Agents who are logged in to the ACD queues handle the interactions.

Inbound Call Flow Examples

This subsection contains several examples of inbound call flows. Each example represents a different outcome.

- An inbound call arrives at an IVR via an ACD queue.
- An inbound call arrives directly at an IVR (see page 107).
- An IVR mute transfers the call to another ACD queue (see page 108).
- An IVR mute transfers the call to an agent (see page 109).

Inbound to IVR via ACD Queue

This call topology shows the outcome of an inbound call to an IVR via an ACD queue. The interaction arrives at the ACD queue and is diverted to an IVR DN.

- Figure 47 depicts the IRF representation of the call topology when the call completes normally in the case of a *self-service* (SS) IVR (when the IVR is in its own box).
- Figure 48 on page 107 depicts the IRF representation of the call topology when the call is abandoned by the customer. This is the *nonself-service* (nonSS) IVR case.

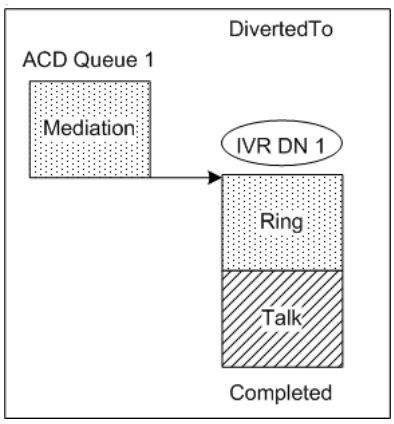

Figure 47: Inbound to IVR via ACD Queue, Completed

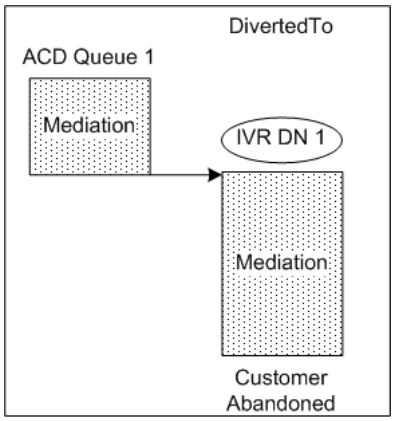

Figure 48: Inbound to IVR via ACD Queue, Abandoned

Inbound to IVR Directly

This call topology shows the outcome of a call that arrives directly at an IVR DN.

- Figure 49 depicts the IRF representation of the call topology when the call completes normally. This is the SS IVR case (when the IVR is in its own box).
- Figure 50 depicts the IRF representation of the call topology when the call is abandoned by the customer. This is the nonSS IVR case.

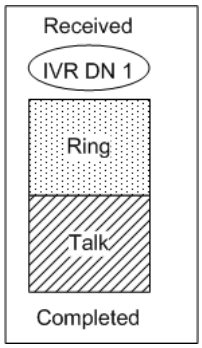

Figure 49: Inbound to IVR Directly, Completed

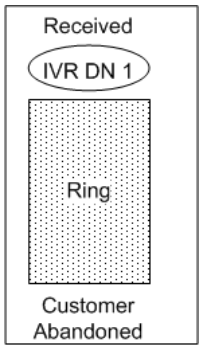

Figure 50: Inbound to IVR Directly, Abandoned

Mute Transfer to ACD Queue

This call topology shows the outcome of a call that is mute transferred to an agent via an ACD queue. The interaction arrives at an ACD queue and is diverted to an IVR DN. The IVR then mute transfers the call to another ACD queue.

- Figure 51 depicts the IRF representation of the call topology in the case of an SS IVR (when the IVR is in its own box).
- Figure 52 on page 109 depicts the IRF representation of the call topology in the case of a nonSS IVR.

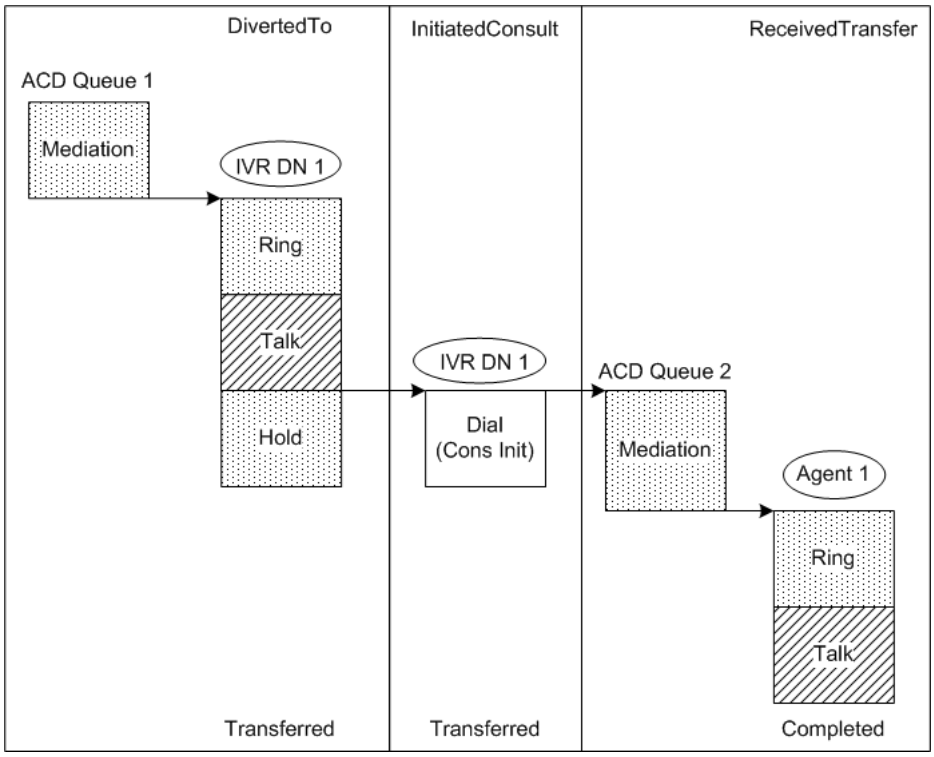

Figure 51: IVR Mute Transfers to ACD Queue (SS IVR)

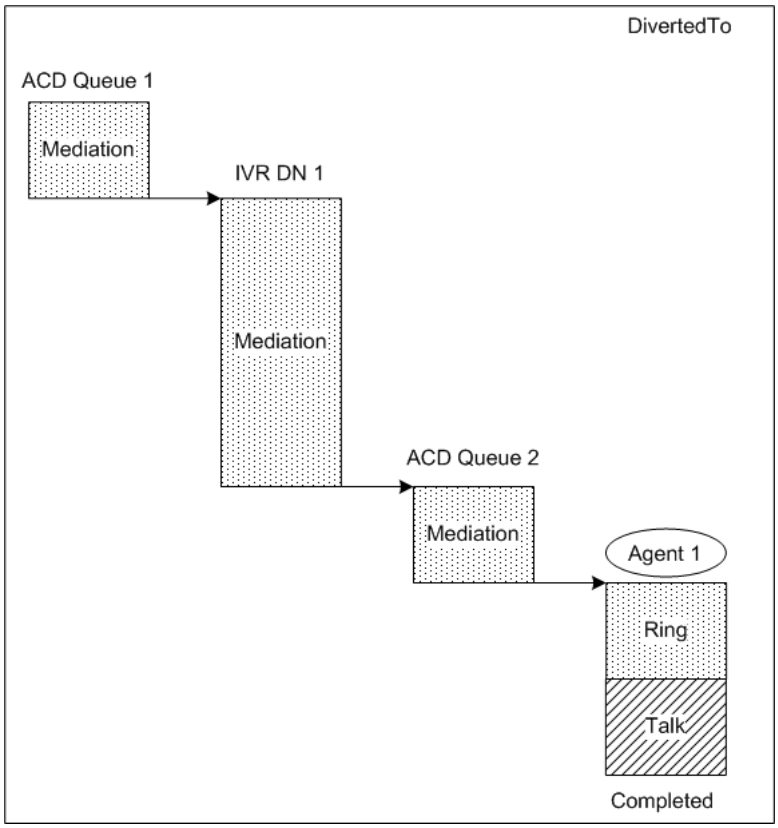

Figure 52: IVR Mute Transfers to ACD Queue (nonSS IVR)

Mute Transfer to Agent

This call topology shows the outcome of a call that is mute transferred to an agent. The interaction arrives at an ACD queue and is diverted to an IVR DN. The IVR then mute transfers the call to an agent.

- Figure 53 on page 110 depicts the IRF representation of the call topology in the case of an SS IVR (when the IVR is in its own box).
- Figure 54 on page 110 depicts the IRF representation of the call topology in the case of a nonSS IVR.

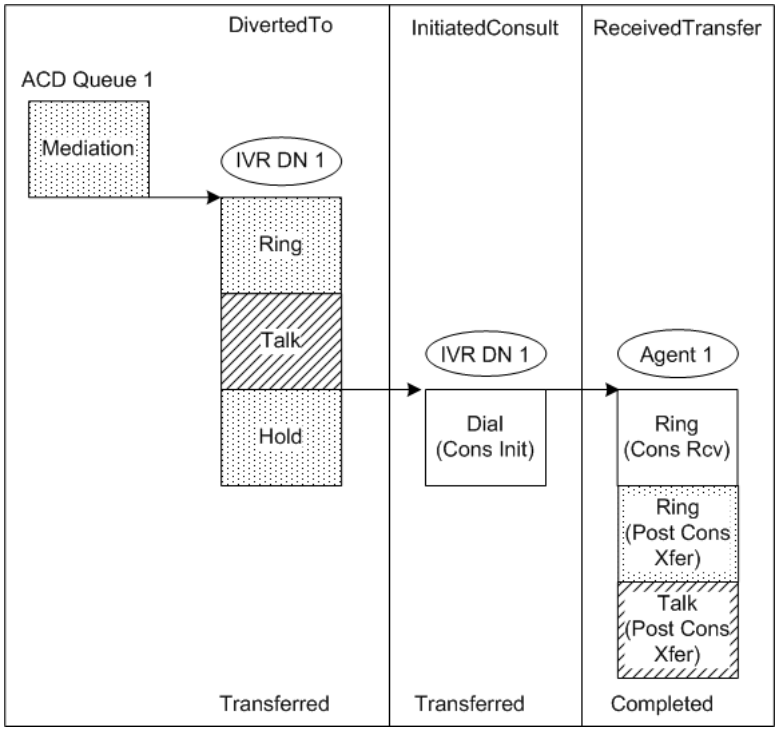

Figure 53: IVR Mute Transfers to Agent (SS IVR)

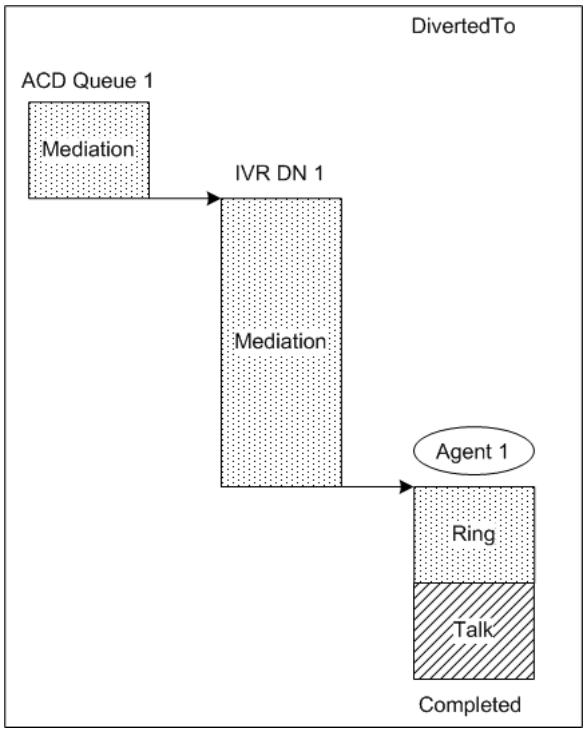

Figure 54: IVR Mute Transfers to Agent (nonSS IVR)

Universal Routing Call Flows

Voice interactions that arrive at the switch are delivered to a routing point. Universal Routing Server uses ANI, DNIS, or the date and time of day to collect information and select an appropriate routing target. Basic targets are ACD queues and individual DNs. More advanced targets are agent groups, place groups, and skill expressions.

Inbound Interactions

This subsection contains the following examples of routed call flows. Each example represents a different outcome:

- A Routing Point routes the call to an ACD queue.
- A Routing Point routes the call to an agent (see page 112).

Routing Point Routes to ACD Queue

This call topology shows the outcome of a call that is routed to an agent via an ACD queue. The call arrives at the Routing Point. The Routing Point then routes the call to an ACD queue, and the interaction is diverted to an agent.

Figure 55 depicts the IRF representation of the call topology.

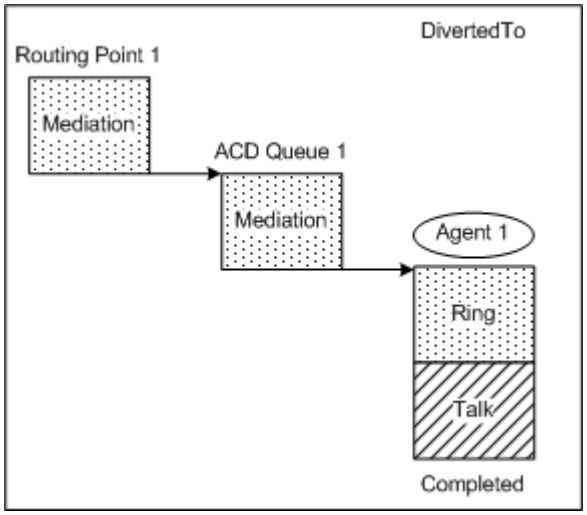

Figure 55: Routing Point Routes to ACD Queue

Note: Figure 55 applies to both network routing and enterprise routing. For network routing, `Routing Point 1` could be a service number on a network T-Server that routes the voice interaction to `ACD Queue 1` on a premise T-Server.

Routing Point Routes to Agent

This call topology shows the outcome of a call that is routed directly to an agent. The call arrives at the Routing Point. The Routing Point then routes the call to an agent.

Figure 56 depicts the IRF representation of the call topology.

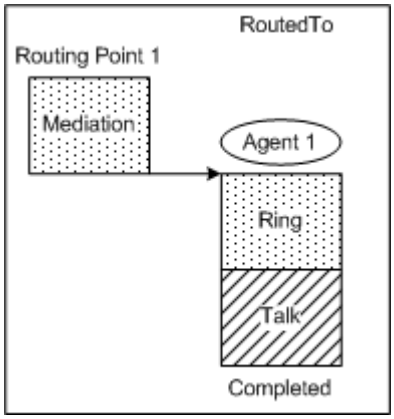

Figure 56: Routing Point Routes to Agent—IRF

Note: Figure 56 applies to both network routing and enterprise routing. For network routing, `Routing Point 1` could be a service number on a network T-Server that routes the voice interaction to Agent 1 on a premise T-Server.

Universal Routing Assisted by IVR-Behind-Switch Call Flows

Voice interactions that arrive at the switch are queued to an ACD queue, where the ACD positions are actually IVR DNs. The IVR collects digits and information about the caller and transfers the call to a Routing Point. Universal Routing uses the collected information to select an appropriate routing target. Basic targets are ACD queues and individual DNs. More advanced targets are agent groups, place groups, and skill expressions.

Inbound Call Flow Examples

This subsection contains several examples of routed call flows. Each example represents a different outcome:

- A Routing Point routes the call to an ACD queue (see page 113).
- A Routing Point routes the call to an agent (see page 114).
- Inbound call to ACD and parallel IVR (see page 116).

Routing Point Routes to ACD Queue

This call topology shows the outcome of a call that is routed to an agent via an ACD queue. The call arrives at an ACD queue and is diverted to an IVR DN. The IVR then transfers the call to a Routing Point, which routes the call to an ACD queue.

- Figure 57 depicts the IRF representation of the call topology in the case of a *self-service* (SS) IVR (when the IVR is in its own box).
- Figure 58 on page 114 depicts the IRF representation of the call topology in the case of a *nonself-service* (nonSS) IVR.

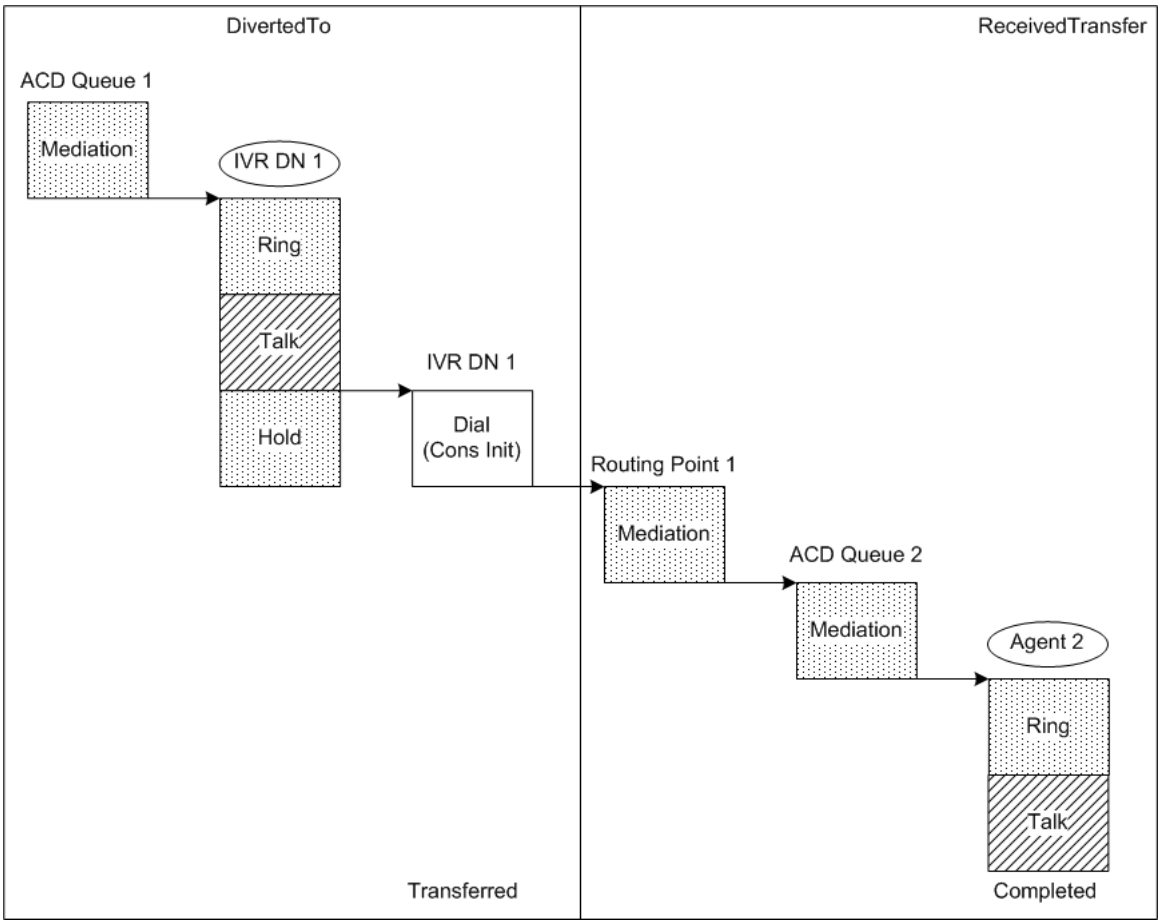

Figure 57: Routing Point Routes to ACD Queue (SS IVR)

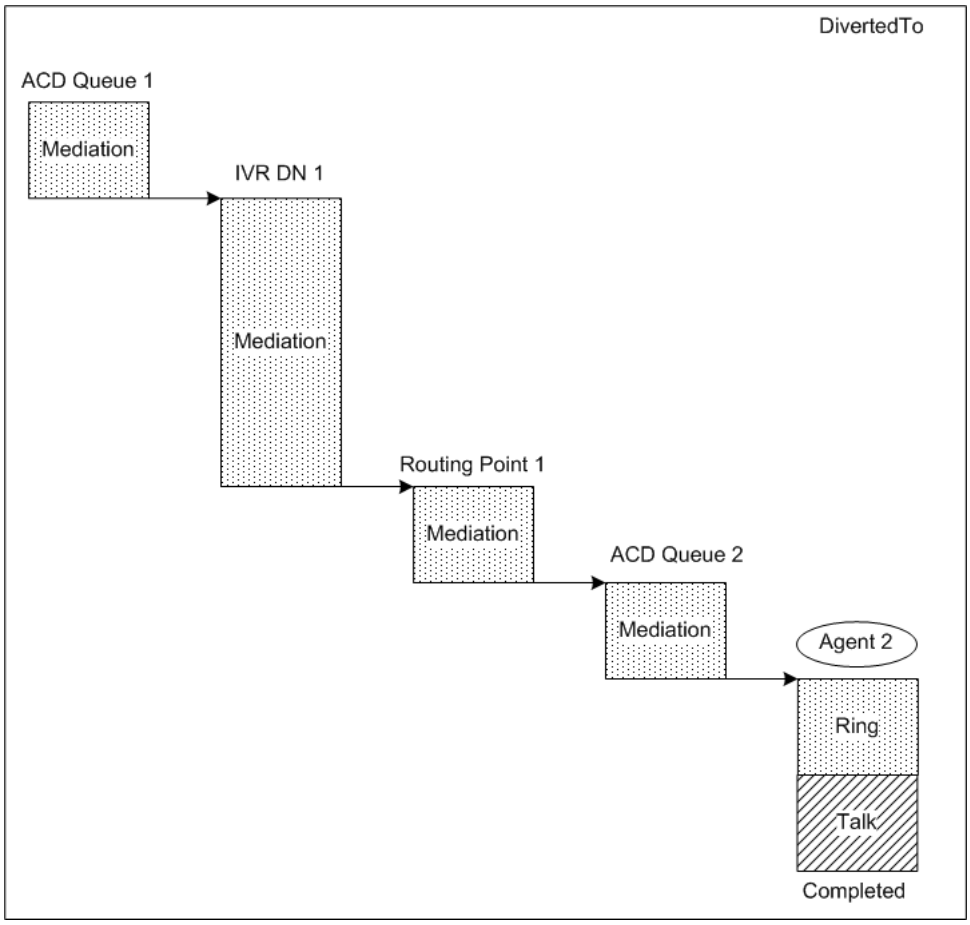

Figure 58: Routing Point Routes to ACD Queue (nonSS IVR)

Routing Point Routes to Agent

This call topology shows the outcome of a call that is routed directly to an agent. The call arrives at an ACD queue and is diverted to an IVR DN. The IVR then transfers the call to a Routing Point, which routes the call to an agent.

- Figure 59 on page 115 depicts the IRF representation of the call topology in the case of an SS IVR (when the IVR is in its own box).
- Figure 60 on page 116 depicts the IRF representation of the call topology in the case of a nonSS IVR.

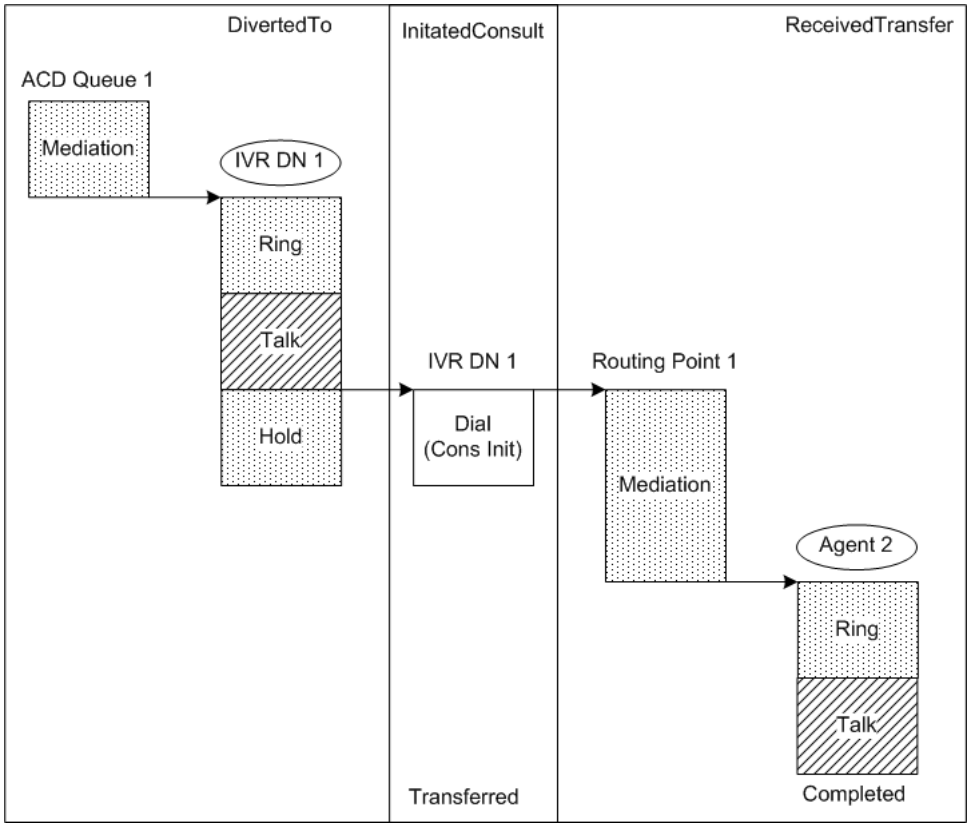

Figure 59: Routing Point Routes to Agent (SS IVR)

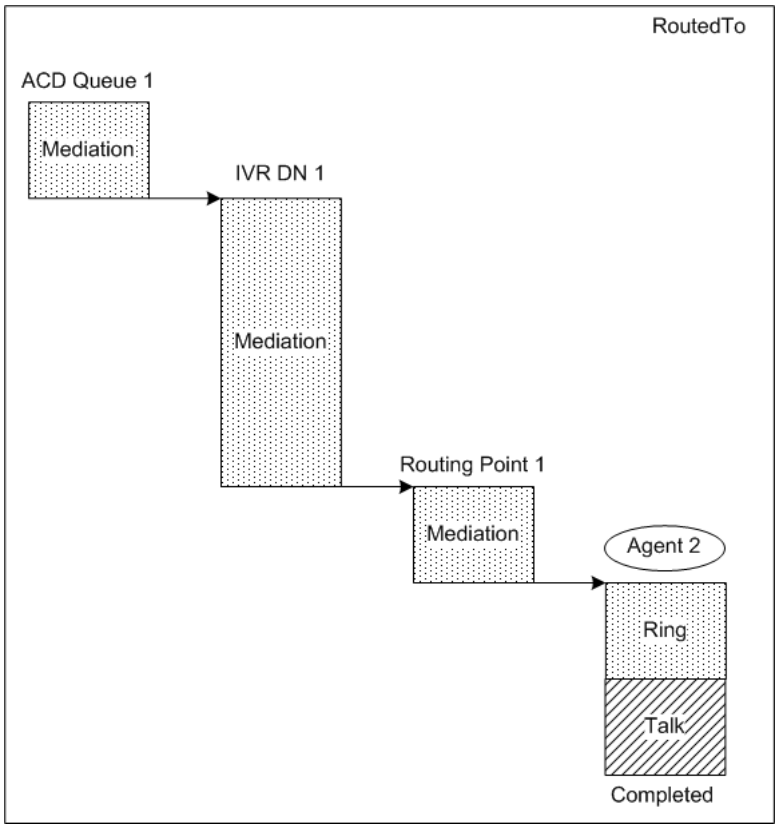

Figure 60: Routing Point Routes to Agent (nonSS IVR)

Inbound Call to ACD and Parallel IVR

This call topology shows the outcome of a call that arrives at a Routing Point, which diverts the call to an ACD queue. While the call is queued at the ACD, the ACD interacts with an IVR, which collects a transfer number from the customer. The IVR returns the collected digits to the ACD, after which the ACD diverts the call to the customer-entered number.

Figure 61 on page 117 depicts the IRF representation of the call topology in the case of a nonSS IVR.

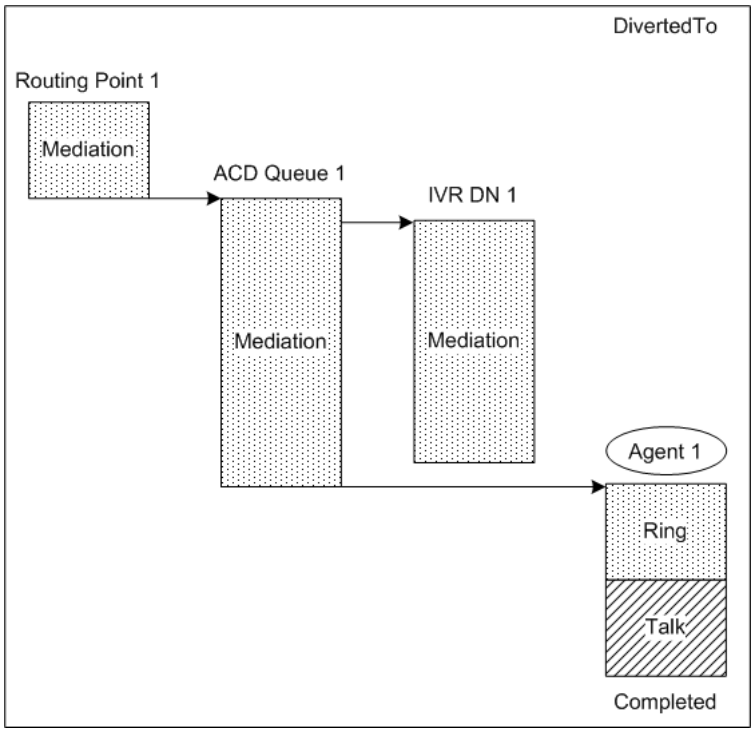

Figure 61: Call is Parallel Queued in ACD Queue and IVR (nonSS IVR)

IVR-in-Front-of-Switch Assisted by Universal Routing Call Flows

Voice interactions arrive at an IVR that is visible to the IVR Server's virtual T-Server. Through a Routing Point in the IVR Server's virtual T-Server, the IVR application invokes an Universal Routing strategy. Universal Routing instructs the IVR application to play applications or collect information. Universal Routing uses the collected information to return an appropriate target. The IVR application hook-flash transfers the call to that target.

Inbound Interactions

This subsection contains several examples of inbound call flows. Each example represents a different outcome:

- The IVR transfers the call to an ACD queue.
- The IVR transfers the call to an agent (see page 118).

IVR Transfers to ACD Queue

This call topology shows the outcome of a call that an IVR transfers to an ACD queue, in accordance with routing instructions. The call arrives at an

IVR DN. The IVR requests routing instructions from a Routing Point, and then hook-flash transfers the call to an ACD queue.

Figure 62 depicts the IRF representation of the call topology.

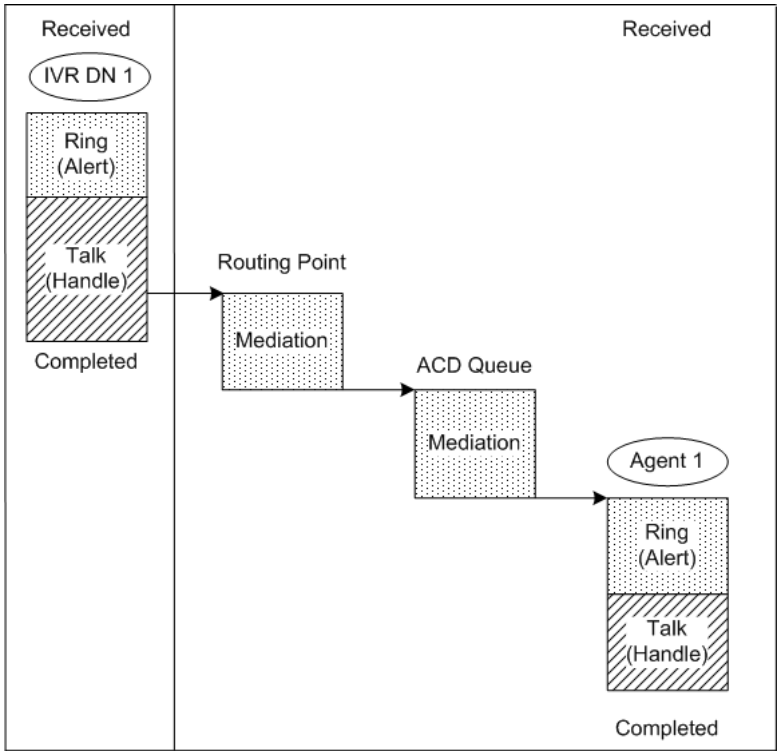

Figure 62: IVR Transfers to ACD Queue

IVR Transfers to Agent

This call topology shows the outcome of a call that an IVR transfers directly to an agent, in accordance with routing instructions. The call arrives at an IVR DN. The IVR requests routing instructions from a Routing Point, and then hook-flash transfers the call to an agent.

Figure 63 on page 119 depicts the IRF representation of the call topology.

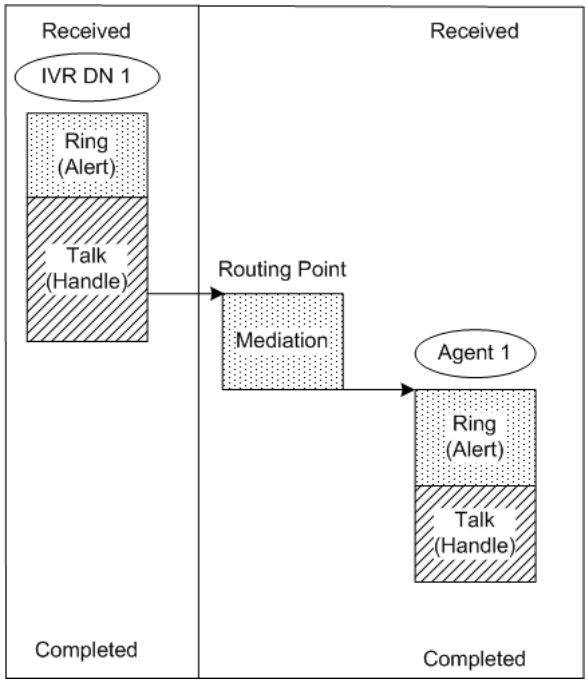

Figure 63: IVR Transfers to Agent

IVR-Behind-Switch Assisted by Universal Routing Call Flows

Voice interactions that arrive at the switch are queued to an ACD queue, in which the ACD positions are actually IVR DNs. Through a virtual routing point in the premise T-Server, the IVR application invokes an Universal Routing strategy. Universal Routing instructs the IVR application to play applications or collect information and uses the collected information to return an appropriate target. The IVR application mute transfers the call to that target.

Inbound Call Flow Examples

This subsection contains the following examples of inbound call flows. Each example represents a different outcome:

- The IVR transfers the call to an ACD queue (see page 119).
- The IVR transfers the call to an agent (see page 121).

IVR Transfers to ACD Queue

This call topology shows the outcome of a call that an IVR transfers to an ACD queue, in accordance with routing instructions. The call arrives at an ACD queue and is diverted to an IVR DN. The IVR requests routing

instructions from a virtual routing point, and then mute transfers the call to another ACD queue.

- Figure 64 depicts the IRF representation of the call topology in the case of a *self-service* (SS) IVR (when the IVR is in its own box).
- Figure 65 on page 121 depicts the IRF representation of the call topology in the case of a *nonself-service* (nonSS) IVR.

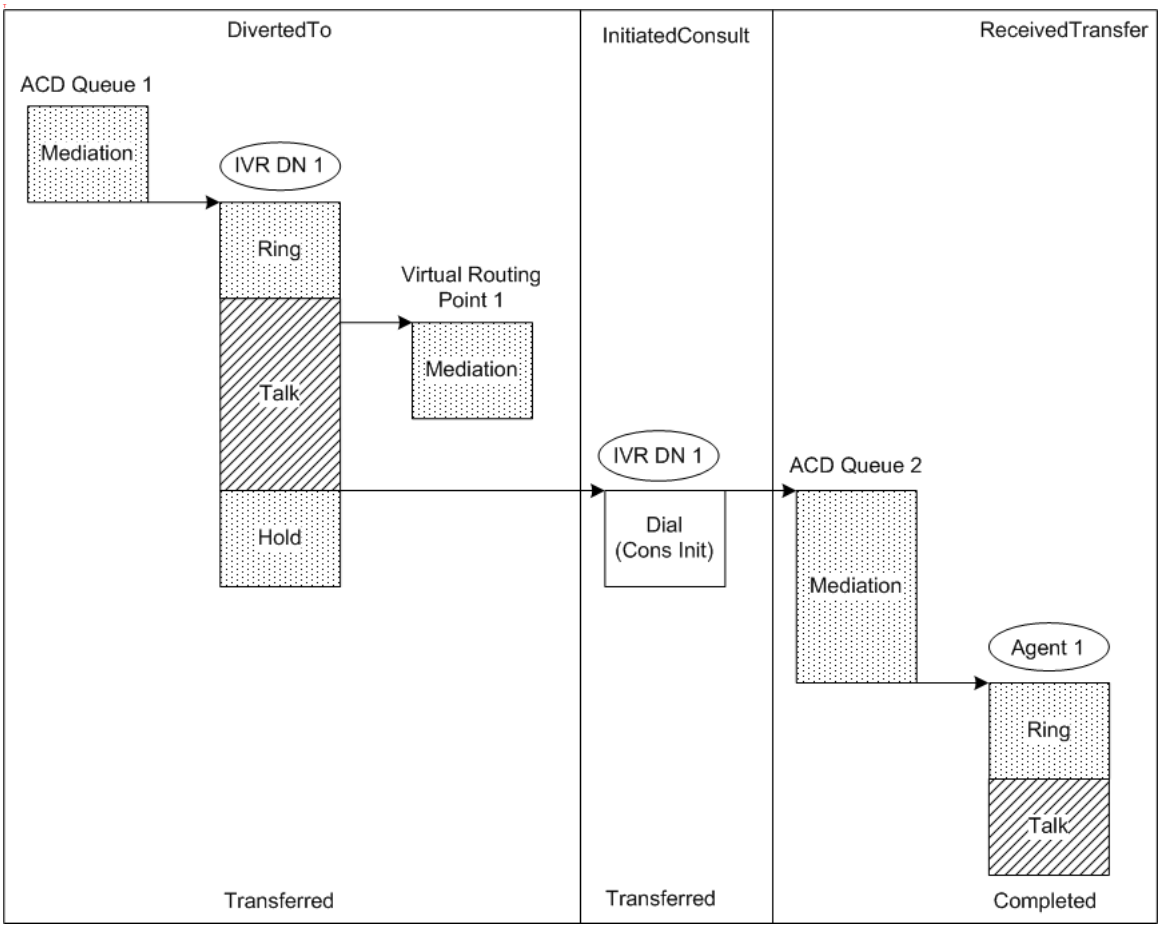

Figure 64: IVR Transfers to ACD Queue (SS IVR)

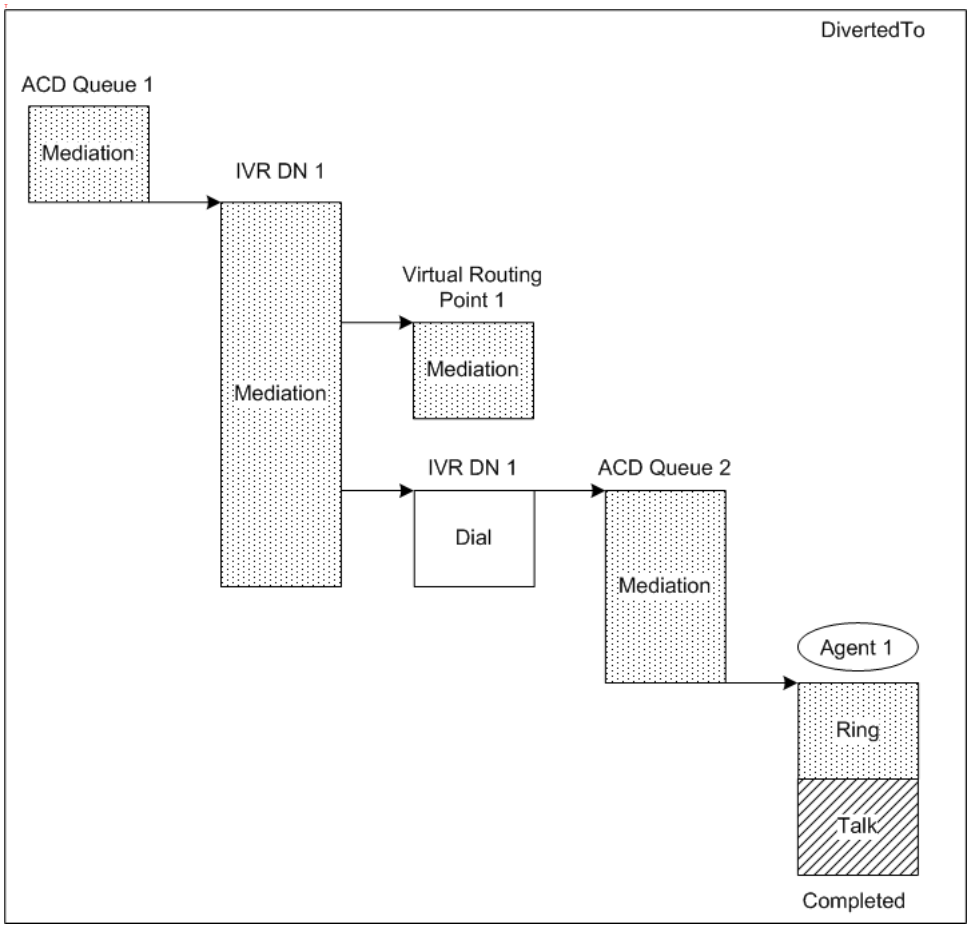

Figure 65: IVR Transfers to ACD Queue (nonSS IVR)

IVR Transfers to Agent

This call topology shows the outcome of a call that an IVR transfers directly to an agent, in accordance with routing instructions. The call arrives at an ACD queue and is diverted to an IVR DN. The IVR requests routing instructions from a virtual routing point, and then mute transfers the call to an agent.

- Figure 66 on page 122 depicts the IRF representation of the call topology in the case of an SS IVR (when the IVR is in its own box).
- Figure 67 on page 123 depicts the IRF representation of the call topology in the case of a nonSS IVR.

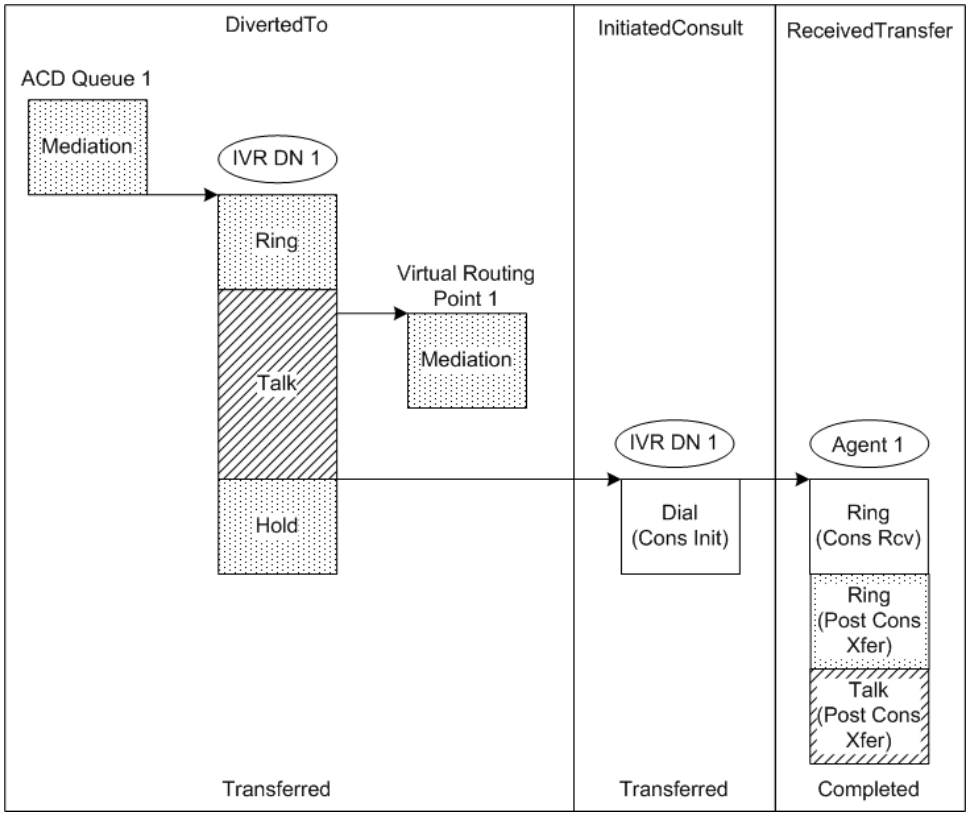

Figure 66: IVR Transfers to Agent (SS IVR)

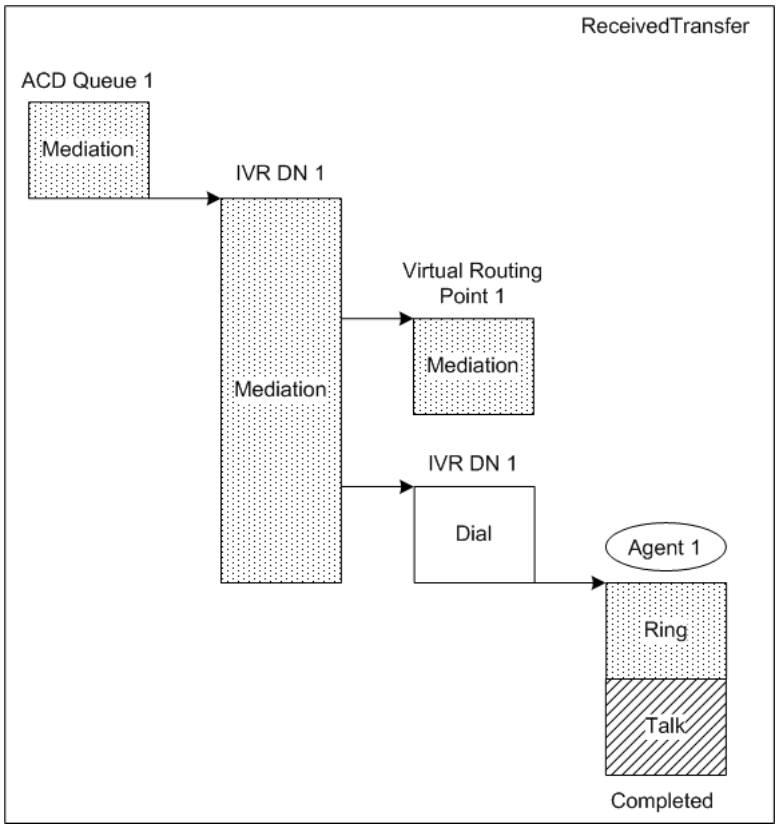

Figure 67: IVR Transfers to Agent (nonSS IVR)

Chapter

5 Validated Multimedia Interaction Flows

This chapter describes the recognized, validated multimedia interactions that have been tested and that are supported by Genesys Info Mart.

This chapter provides detailed sections that discuss Genesys eServices e-mail and chat. However, Genesys Info Mart supports full processing of any 3rd Party Media interactions, in addition to e-mail and chat interactions.

Use the sections that discuss e-mail as a guide to interactions that do not involve an online session with a customer (offline interactions), and the chat section as a guide to interactions that do involve an online session with a customer (online interactions).

The interaction flows described in this chapter are intended as examples that you can modify for your environment. However, Genesys does not guarantee results for modified interaction flows.

This chapter contains the following sections:

Diagram Conventions

Like the voice interaction flows in Chapter 4 on page 65, the flow diagrams in this chapter represent the resources that participate in interactions and the resources' states. The diagrams use the same conventions as the call flow diagrams in Chapter 4 (see "Diagram Conventions" on page 67, including "Notes on the Interaction-Flow Diagrams" on page 69).

Additional Conventions

The following additional features are significant for the multimedia interaction-flow diagrams:

- Terms such as *Ring* and *Talk* are used generically. In multimedia interaction flows, these terms indicate the *Alerting* and *Connected* conditions.
- For both online and offline interactions, the customer is considered to be present ("talking") during the whole interaction, except for any consultation portions.
- The resources of interest are handling resources, which are the resources that have the greatest interest for reporting. Primarily, these are agents, but routing strategies that send an Autoresponse are also considered to be handling resources and become the subjects of IRF records. Additionally, when an interaction is abandoned in an Interaction Queue or a Routing Strategy, the resource in which it was abandoned is represented with an IRF entry.
- Strategies are not mediation resources. The diagrams illustrate the use of virtual queues (VQs) to enable time that an interaction spends in a strategy to be reported in a Mediation Segment Fact (MSF) record, as mediation time. If no VQ is defined in a strategy, there will be gaps in the reported mediation time.
- In all cases in which a routing strategy routes an interaction to an agent, the diagrams show the interaction flow beginning in mediation, while the interaction waits in Interaction Queue 1. When Strategy 1 attempts to find a routing target, the strategy keeps the interaction in VQ1 (another mediation). Figure 75 on page 137 shows the effect of more complicated routing strategies in which there are multiple parallel and sequential VQs. On the other hand, all the diagrams that end with the e-mail being sent out of the contact center show a strategy that is not being used for routing purposes and, therefore, for which VQs do not apply.
- The diagrams show VQ time and strategy time as identical, which is always the case when `adjust-vq-time-by-strategy-time=true`. This means that mediation duration includes time that the interaction spent in the strategy but outside the VQ, and there are no gaps in the mediation time. For more information about the option, see the *Genesys Info Mart 8.1 Deployment Guide*.
- Except where otherwise indicated (for example, in Figure 70 on page 130), the diagrams assume that Genesys Info Mart has been configured to populate Interaction Queue activity in the MSF table (`populate-mm-ixnqueue-facts=true`). MSFs for Interaction Queue activity have been included in the diagrams for completeness. By default, `populate-mm-ixnqueue-facts=false`. For more information about the option, see the *Genesys Info Mart 8.1 Deployment Guide*.
- Given the focus and purpose of this chapter, the e-mail diagrams for interaction flows that end when the e-mail is sent out of the contact center show the e-mail being sent to an Interaction Queue (named `Outbound Queue`), from which a strategy (named `Outbound Strategy`) sends the e-mail to the customer. However, more complicated scenarios are possible, in

which case the reported technical result might be different. For example, if the outbound e-mail is first sent to a supervisor for a quality review, there will be additional mediation, and the technical result in the IRF for the handling agent will be `Transferred` instead of `Completed`.

E-Mail Interactions

This section contains several examples of e-mail flows. Each example represents a different outcome:

- A routing strategy routes the e-mail interaction to an agent, and the agent replies (see page 128).
- A routing strategy routes the e-mail interaction to an agent, but the agent does not accept the invitation (see page 129).
- An incoming e-mail interaction is handled by a routing strategy with an autoresponse (see page 129).
- A routing strategy routes the e-mail interaction to an agent, who transfers it directly to another agent (see page 130).
- A routing strategy routes the e-mail interaction to an agent, who unsuccessfully attempts to transfer it directly to another agent (see page 131).
- A routing strategy routes the e-mail interaction to an agent, who transfers the e-mail through a queue to another agent, who replies to the e-mail (see page 132).
- An agent unsuccessfully attempts to transfer an e-mail interaction to another agent through a queue (see page 134).
- A routing strategy routes the e-mail interaction to an agent, who transfers the e-mail through a queue to another agent, when the routing strategies contain multiple VQs (see page 136).
- A routing strategy routes the e-mail interaction to an agent, who consults to another agent before sending a reply and continues to work on the reply during the consultation period (see page 138).
- A routing strategy routes the e-mail interaction to an agent, who consults to another agent before sending a reply and does not continue to work on the reply during the consultation period (see page 140).
- A routing strategy routes the e-mail interaction to an agent, who unsuccessfully attempts to consult to another agent before sending a reply (see page 142).
- An agent saves a draft of the e-mail reply (workbin time is considered to be mediation), and then later completes and sends it (see page 143).
- An agent pulls the e-mail interaction from an interaction queue or workbin (see page 145).

- An agent saves a draft of the e-mail reply (workbin time is considered to be hold), and then later completes and sends it (see page 145).
- Multipart reply (see page 146).
- Agent(s) create multiple replies to an inbound e-mail, one of which is stopped without being sent (see page 148).
- A routing strategy eventually routes an e-mail interaction to an agent, after repeatedly failing to find an available agent (see page 150).

Strategy Routes E-Mail to Agent, and Agent Replies

Figure 68 shows the outcome of an e-mail interaction that a routing strategy routes to Agent 1, who accepts the invitation (IRF1). Agent 1 creates an outbound reply e-mail (IRF2), closing the original inbound e-mail. The outbound e-mail is placed into Outbound Queue, from which Outbound Strategy sends it out of the contact center to the customer.

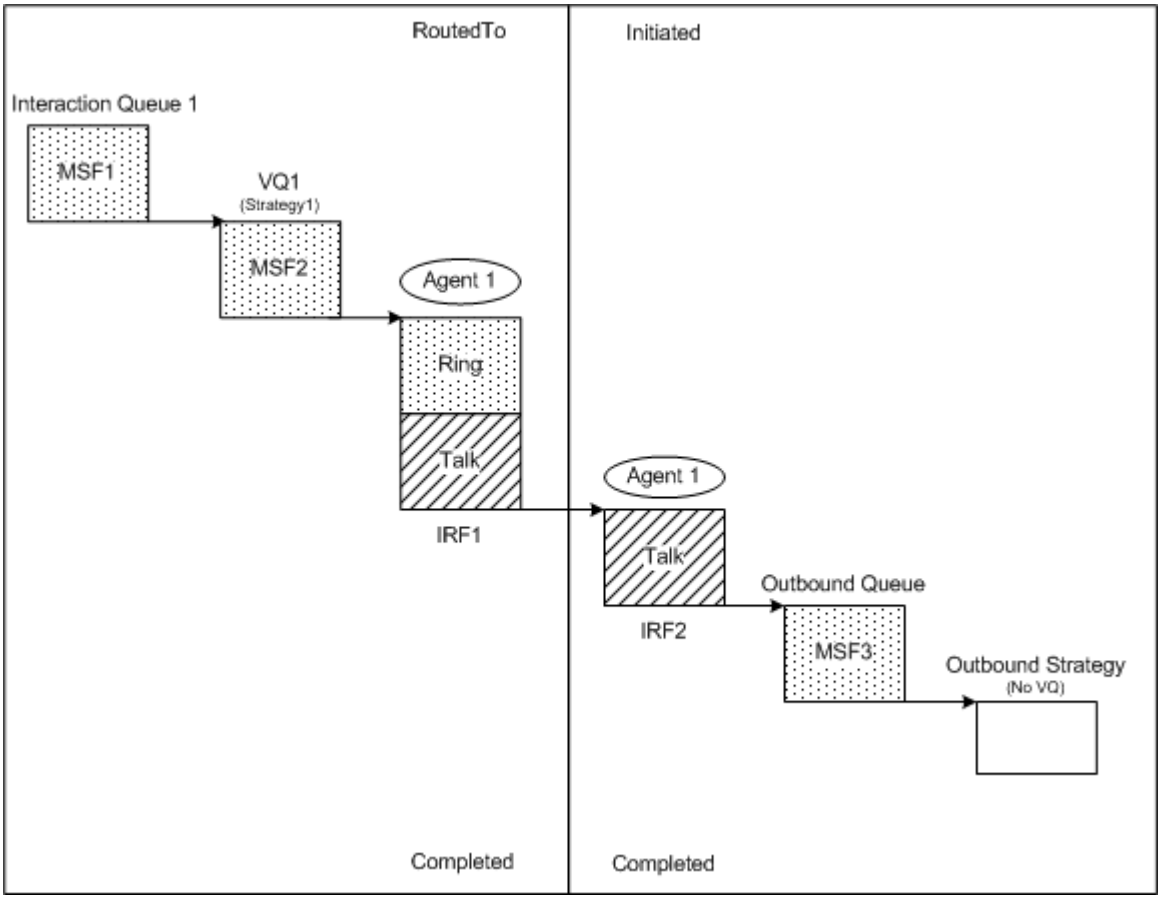

Figure 68: Strategy Routes E-Mail to Agent, and Agent Replies

Agent Invited into E-Mail Interaction, and Invitation Revoked

Figure 69 shows the outcome when an e-mail interaction is offered to an agent by a routing strategy, but the agent does not accept the invitation. The e-mail interaction is returned to the interaction queue so that it can be re-evaluated.

Note: When a routing strategy routes to an agent, the strategy is removed from the interaction as soon as the agent is invited into that interaction. In other words, the routing is complete as soon as the agent is invited.

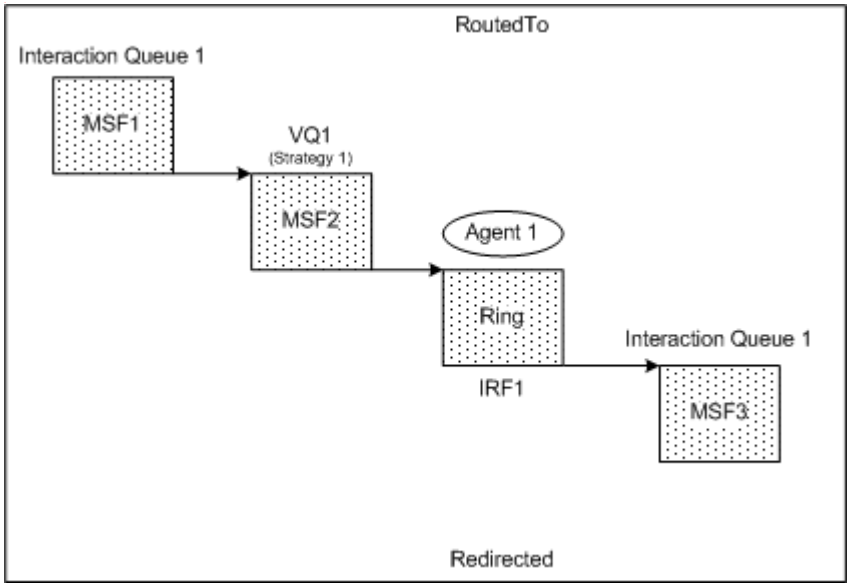

Figure 69: Agent Does Not Accept E-Mail Invitation

E-Mail Interaction Handled by a Strategy with Autoresponse

Figure 70 on page 130 shows the outcome of an e-mail interaction that a routing strategy determines can be handled with an Autoresponse. The e-mail is submitted to an inbound interaction queue. The routing strategy pulls the e-mail from the interaction queue and identifies that it can be handled with an Autoresponse, and an Autoresponse is generated. Strategy 1 is now considered to be a handling resource. Strategy 1 connected to and stopped the original inbound interaction, represented in IRF1, and created an outbound Autoresponse reply, which is represented in IRF2. The time that Strategy 1 is connected to each e-mail is represented as “Talk” time. Outbound Queue and Outbound Strategy represent the processing that occurs when the e-mail is sent outside the contact center.

This example assumes that `populate-mm-ixnqueue-facts=false`, so there are no mediations to report.

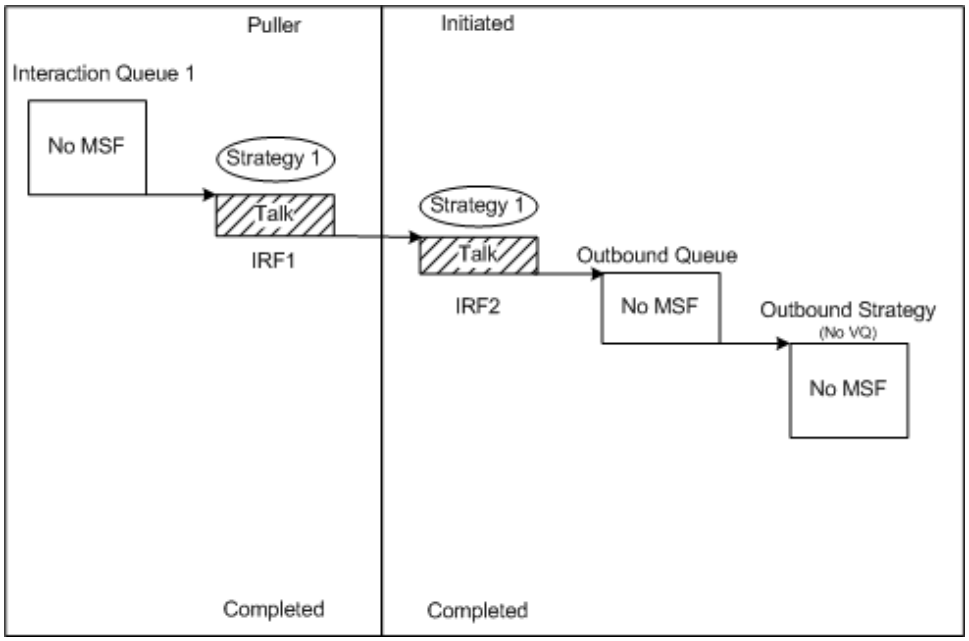

Figure 70: E-Mail Handled by a Strategy with Autoresponse

Agent Transfers E-Mail Directly to Another Agent

Figure 71 on page 131 shows the outcome of an e-mail interaction that is routed to an agent, who transfers the e-mail to another agent, who replies to the e-mail.

Agent 1 (IRF1) transfers an inbound interaction to Agent 2. Agent 2 stops the original inbound interaction while creating an outbound reply (IRF2 and IRF3). The outbound reply is placed into Outbound Queue, from which Outbound Strategy sends it out of the contact center to the customer.

When an agent directly invites another agent into an interaction, the original agent remains in the interaction until the target agent accepts the invitation. In the case of a transfer, the transfer does not occur until the target agent accepts the invitation.

Note: In this scenario, the original inbound e-mail is transferred. Figure 72 on page 132 presents a partial variation on this scenario, in which Agent 1 creates an outbound reply and attempts to transfer the reply to Agent 2.

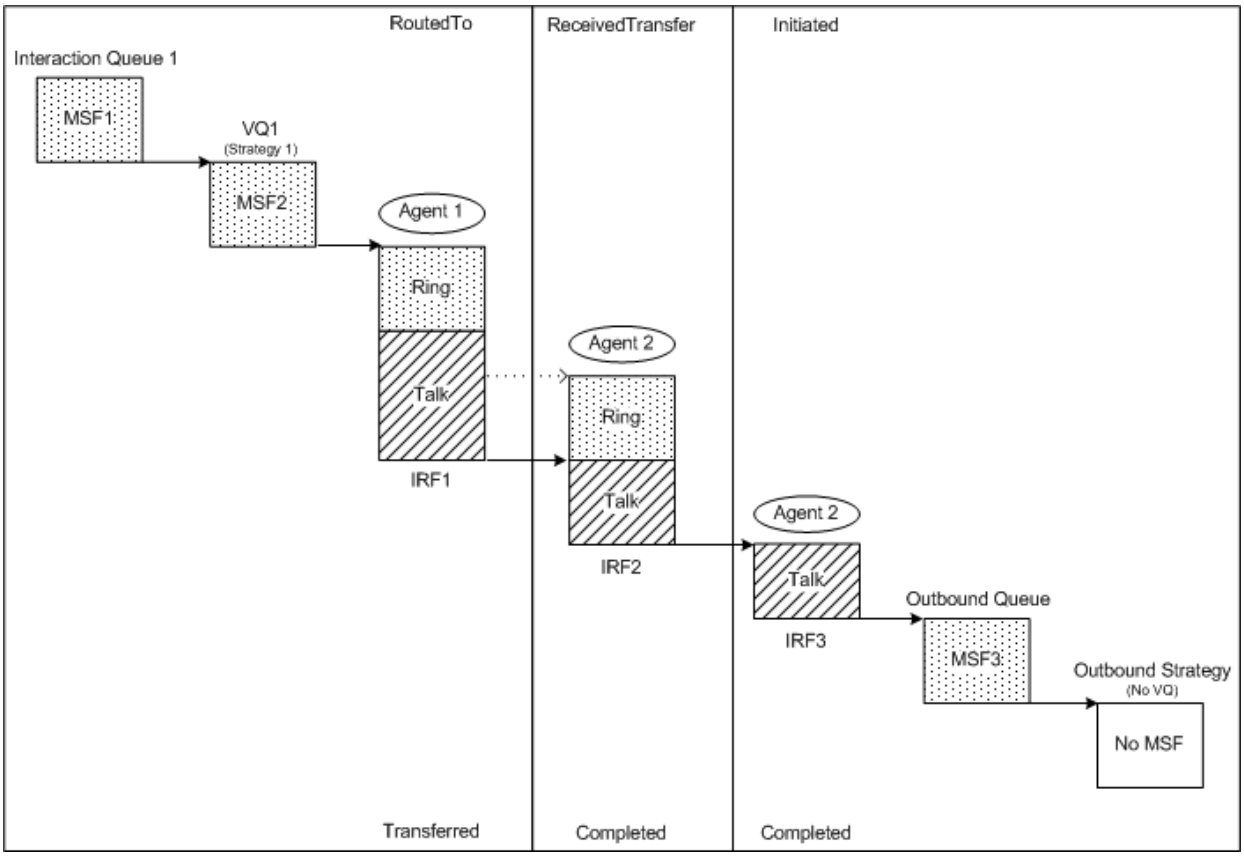

Figure 71: Agent Transfers E-Mail to Another Agent

Agent's Attempt to Transfer E-Mail Directly to Another Agent Fails

Figure 72 on page 132 shows the outcome of an unsuccessful attempt to transfer an e-mail to another agent.

The interaction is routed to Agent 1. Agent 1 accepts the inbound e-mail (IRF1) and creates an outbound reply (IRF2), closing the original inbound e-mail. Agent 1 works on this reply and then attempts to transfer this reply to Agent 2, for Agent 2 to complete the reply (IRF3). Agent 2 does not accept the invitation into the interaction. Agent 1 remains in the interaction during the attempt to transfer. In this case, since Agent 2 was not available, Agent 1 completes the reply and places it into Outbound Queue, from which Outbound Strategy sends it out of the contact center to the customer.

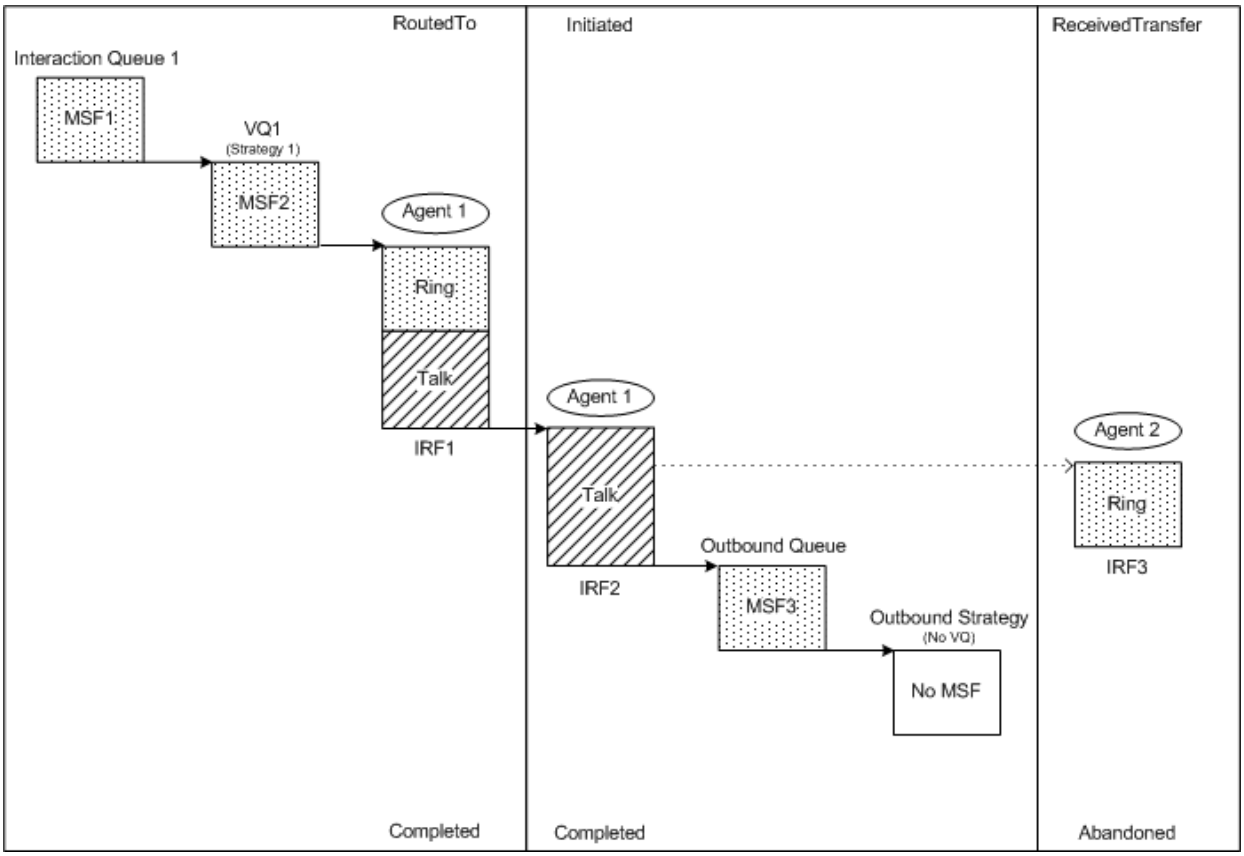

Figure 72: Unsuccessful Direct Agent-to-Agent E-Mail Transfer Attempt

Transfer of E-Mail from Agent to Agent through a Queue

Figure 73 on page 133 shows the outcome of an e-mail interaction that is routed to an agent, who transfers the e-mail through a queue to another agent, who replies to the e-mail.

Agent 1 transfers an inbound interaction through Interaction Queue 2 to Agent 2 (IRF1). Agent 2 stops the original inbound interaction while creating an outbound reply (IRF2 and IRF3). The outbound reply is placed into Outbound Queue, from which Outbound Strategy sends it out of the contact center to the customer.

Note: In this scenario, the original inbound e-mail is transferred. Figure 72 on page 132 presents a partial variation on this scenario, in which Agent 1 creates an outbound reply and attempts to transfer the reply to Agent 2.

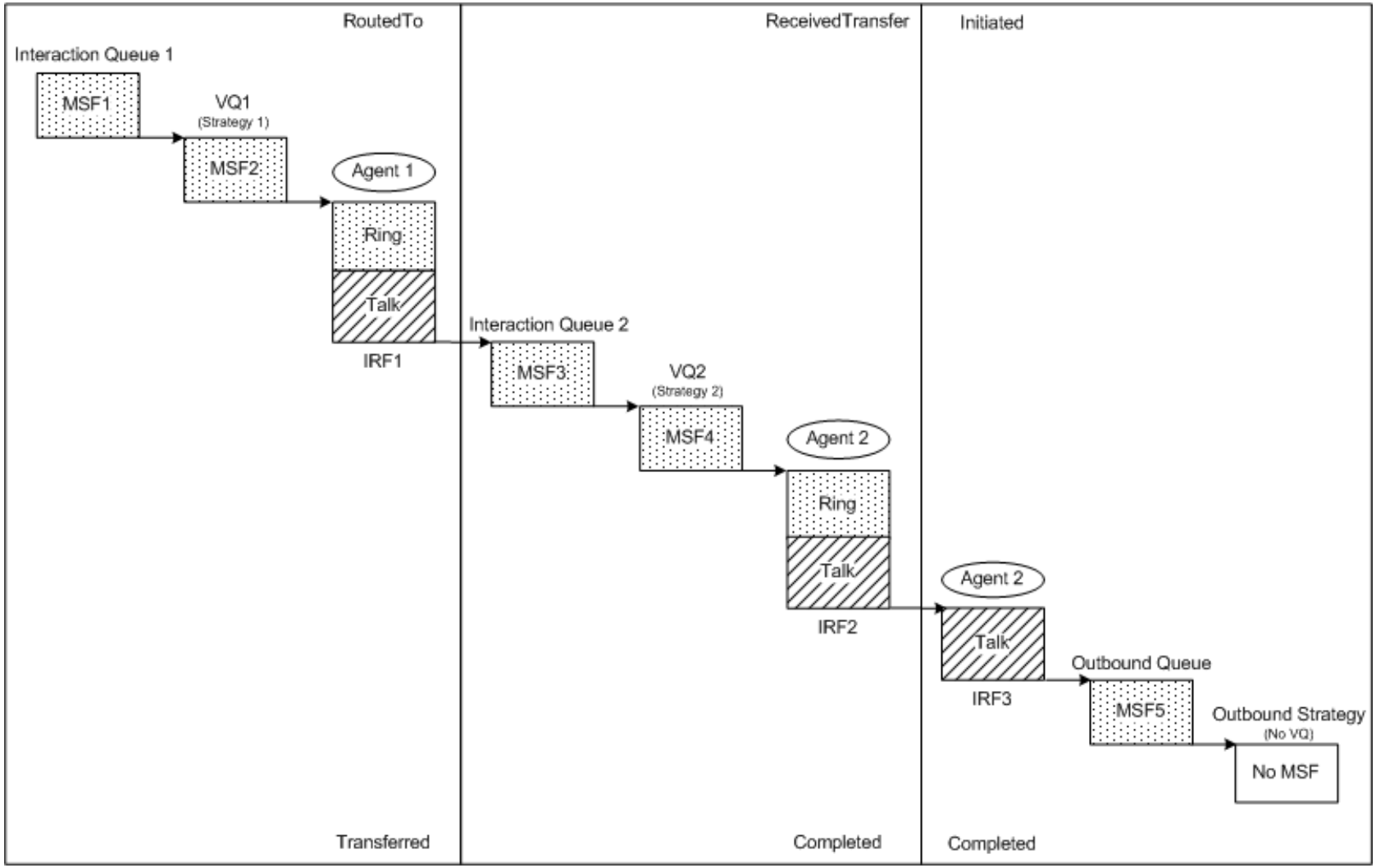

Figure 73: Transfer of E-Mail from Agent to Agent through a Queue

Unsuccessful Transfer from Agent to Agent through a Queue

Figure 74 on page 135 shows the outcome of an unsuccessful attempt to transfer an e-mail interaction from one agent to another agent through a queue.

The interaction is routed to Agent 1. Agent 1 accepts the inbound e-mail (IRF1) and creates an outbound reply, closing the original inbound e-mail (IRF2). Agent 1 works on this reply, and then attempts to transfer the reply interaction through Interaction Queue 2 to another agent for continued processing. The reply interaction is routed from the queue to Agent 2 (IRF3). Agent 2 does not accept the invitation, and this revoked invitation is returned to Interaction Queue 2, so that another routing target can be found.

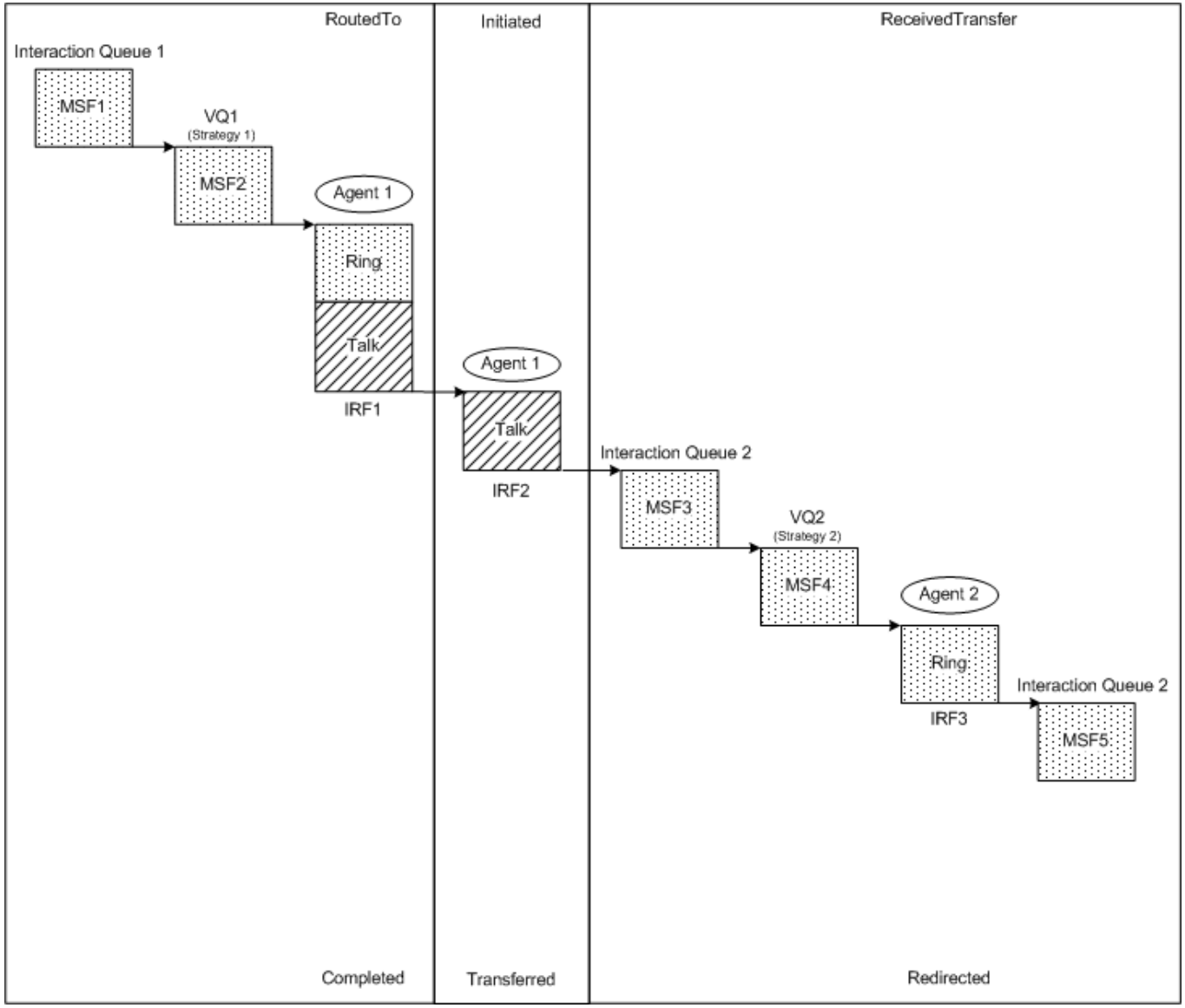

Figure 74: Unsuccessful Transfer from Agent to Agent through a Queue

Transfer from Agent to Agent through a Queue with Multiple VQs

Figure 73 on page 133 shows the outcome of an inbound e-mail interaction that is routed to an agent, who transfers the e-mail through a queue to another agent, when the routing strategies contain several VQs. In this example, Strategy 1 contains three parallel VQs (VQ1, VQ2, VQ3) and Strategy 2 contains three sequential VQs (VQ4, VQ5, VQ6).

Strategy 1 takes the e-mail from Interaction Queue 1 and places the e-mail simultaneously in all three VQs that are associated with the strategy. Strategy1 finds a routing target through VQ1 and routes the interaction to the target (Agent 1).

Agent 1 handles the e-mail (IRF1) and then decides to send it to another agent through Interaction Queue 2, for continued processing. Strategy 2 takes the e-mail from Interaction Queue 2 and places it in VQ4. When Strategy 2 does not find an available agent who is associated with VQ4, the strategy places the e-mail in VQ5, where it finds an available agent (Agent 2). The strategy routes the e-mail to Agent 2 (IRF2). Because the interaction did not reach the third VQ that is associated with Strategy 2, there is no MSF for VQ6. Given the focus of this example, the diagram does not show the continued processing after Agent 2 gets control of the interaction; the technical result for IRF2 depends on what Agent 2 does with the interaction.

This example assumes that `populate-mm-ixnqueue-facts=false`, so there are no mediations to report for the Interaction Queues.

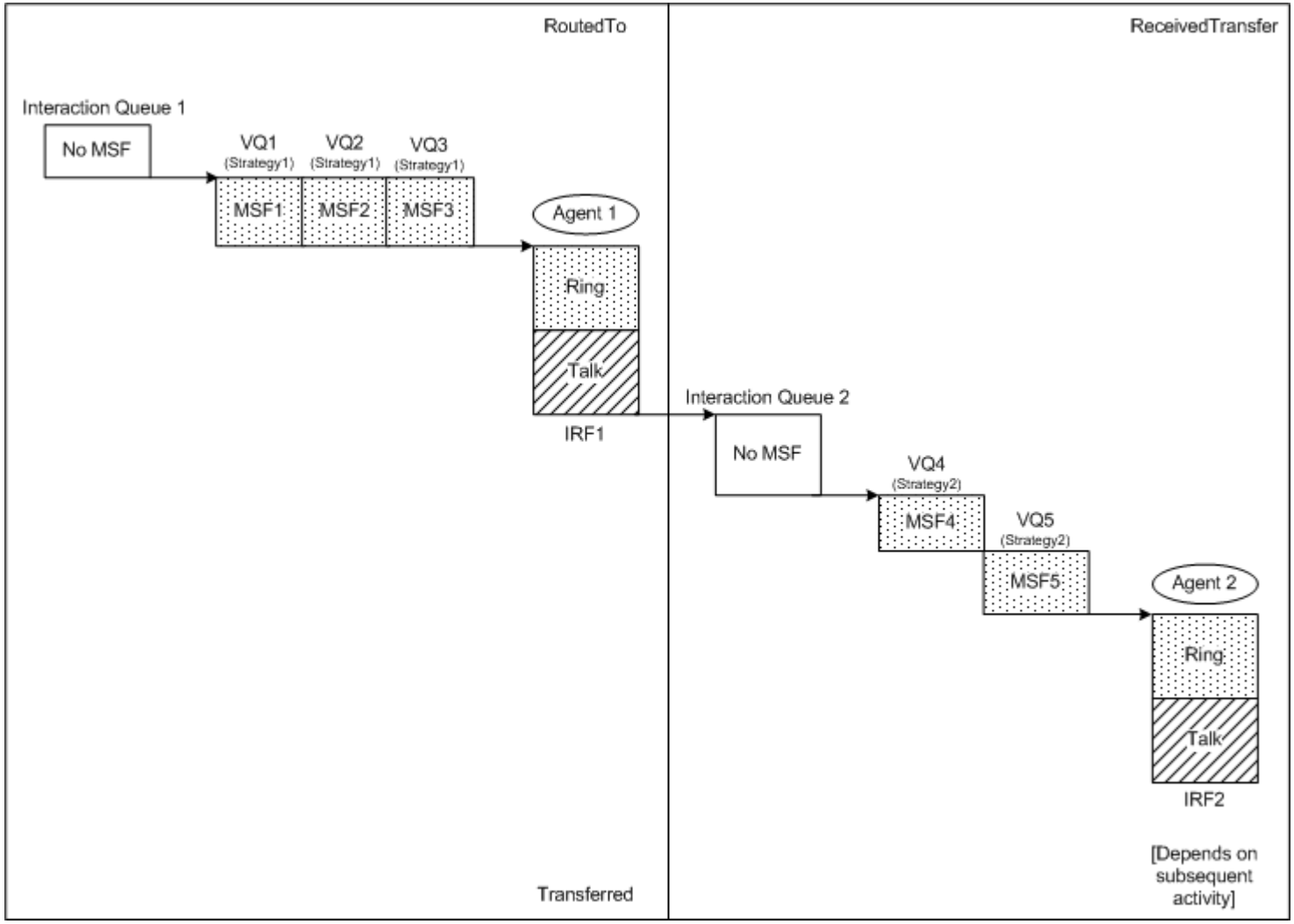

Figure 75: Transfer from Agent to Agent with Parallel and Sequential VQs

Agent Consults to Another Agent Before Sending Reply—Non-Blocking

Figure 76 on page 139 shows an agent consulting with another agent before sending a reply, when the first agent continues to work on the reply during the consultation period.

The interaction is routed to Agent 1. Agent 1 accepts the inbound e-mail (IRF1) and creates an outbound reply, closing the original inbound e-mail (IRF2). Agent 1 initiates a Consult interaction and transfers it to Agent 2 (IRF3). Agent 2 accepts the Consult interaction (IRF4) and initiates a Consult Reply interaction (IRF5). In a typical Consult scenario, the Consult Reply is placed into Agent 1's Collaboration Workbin. The Consult Reply typically remains in the workbin for the life of the entire interaction, enabling the Consult Reply to be viewed at any time during the processing of the interaction and making it available for viewing by another agent, if Agent 1 transfers ownership of this interaction to another agent.

The outbound reply is placed into Outbound Queue, from which Outbound Strategy sends it out of the contact center to the customer.

Notes:

- By placing the Consult Reply in Agent 1's Collaboration Workbin, Agent 2 transfers the Consult Reply back to Agent 1. In a common scenario, the Consult Reply remains in the Collaboration Workbin for the remainder of the interaction, until it is cleaned up when the interaction ends. At that time, a new IRF row (not shown) is added, to show the Consult Reply being pulled from the workbin and `Completed`.
- The "Talk" boxes in the Consult portion of the diagram are not shaded, because the customer is not considered to be present during offline consultations.

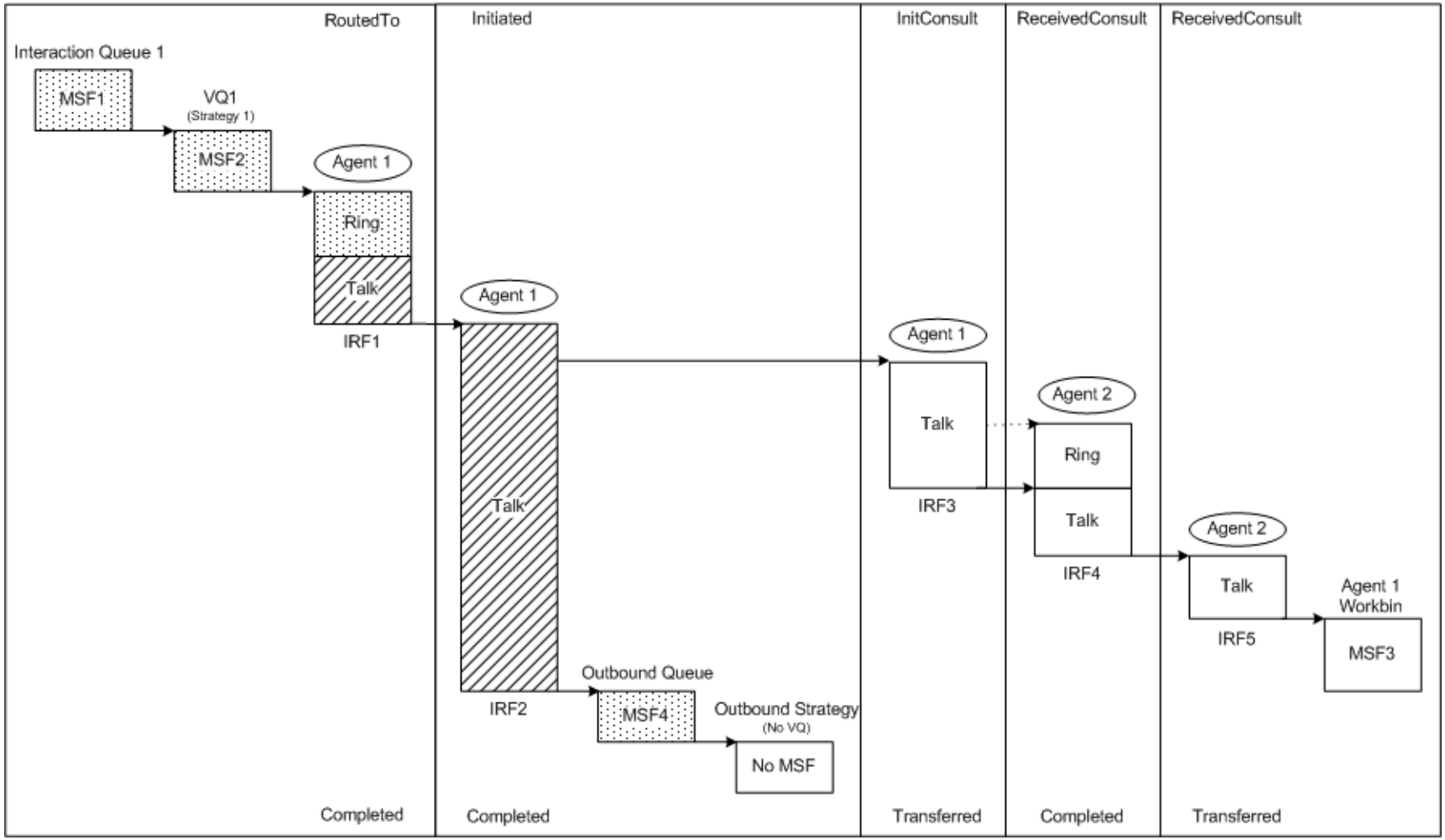

Figure 76: Agent Consults to Another Agent Before Sending Reply—Non-Blocking Consultation

Agent Consults to Another Agent Before Sending Reply—Blocking

Figure 77 on page 141 shows an agent consulting with another agent before sending a reply, when the first agent suspends work on the reply until the consultation response is received.

The interaction is routed to Agent 1. Agent 1 accepts the inbound e-mail (IRF1) and creates an outbound reply, closing the original inbound e-mail (IRF2). Agent 1 initiates a Consult interaction and transfers it to Agent 2 (IRF3). In the meantime, Agent 1 suspends work on the outbound reply and places it in a Draft Workbin (MSF3), until the results of Agent 2's collaboration are available. Agent 2 accepts the Consult interaction (IRF4) and initiates a Consult Reply interaction (IRF5). After working on the Consult Reply, Agent 2 places it in Agent 1's Collaboration Workbin (MSF4), where it typically remains for the remaining life of the interaction. Agent 1 retrieves the suspended outbound reply from the Draft Workbin and completes the reply (IRF6).

The outbound reply is placed into Outbound Queue, from which Outbound Strategy sends it out of the contact center to the customer.

Notes:

- For the time that the interaction is in Agent 1's Draft Workbin, the diagram illustrates the situation when workbin time is considered to be mediation (`populate-workbin-as-hold =false`). Figure 81 on page 146 shows the flow for this portion of the interaction when the workbin time is considered to be hold.
- By placing the Consult Reply in Agent 1's Collaboration Workbin, Agent 2 transfers the Consult Reply back to Agent 1. In a common scenario, the Consult Reply remains in the Collaboration Workbin for the remainder of the interaction, until it is cleaned up when the interaction ends. At that time, a new IRF row (not shown) is added, to show the Consult Reply being pulled from the workbin and `Completed`.
- The "Talk" boxes in the Consult portion of the diagram are not shaded, because the customer is not considered to be present during offline consultations.

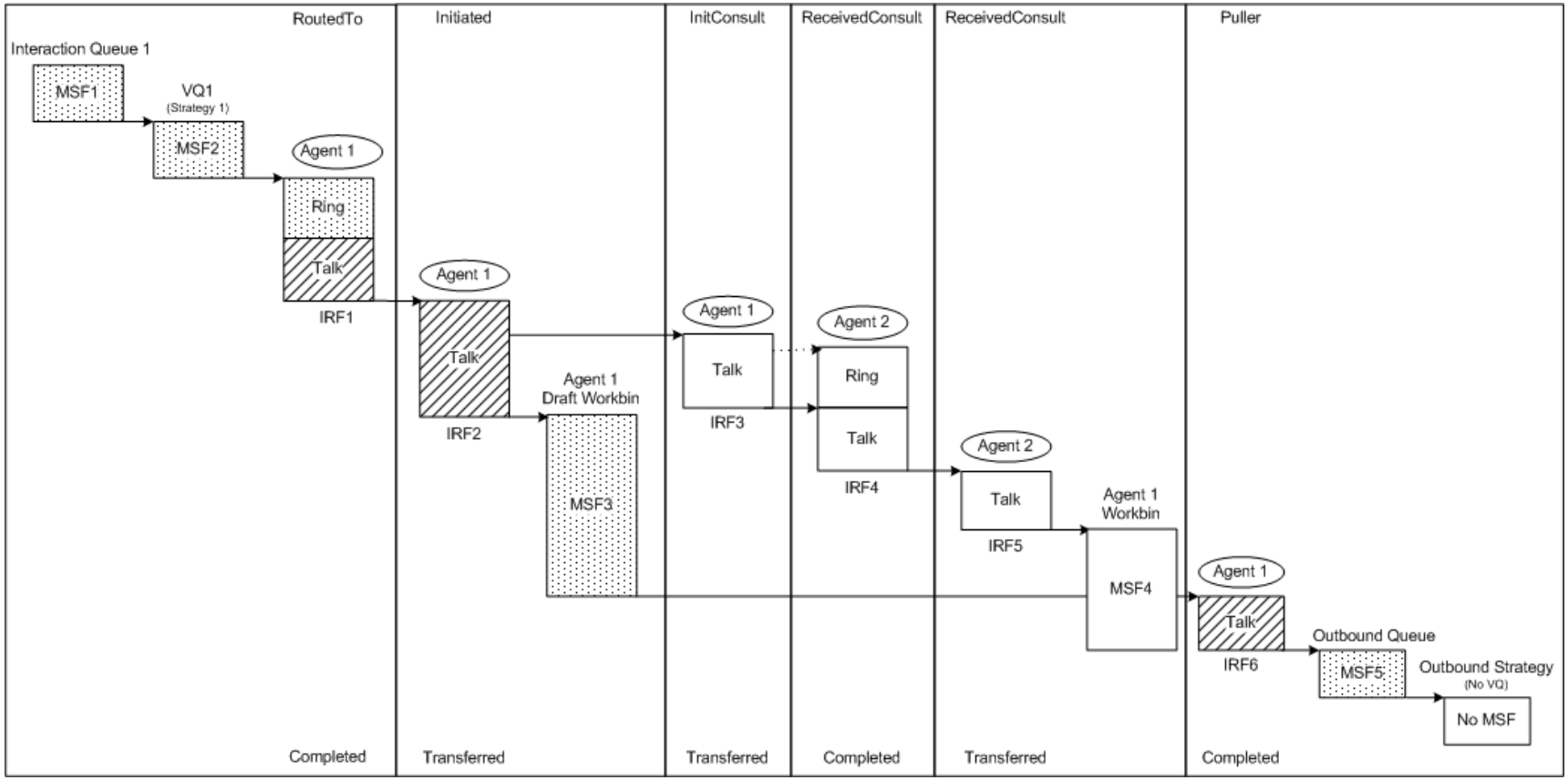

Figure 77: Agent Consults to Another Agent Before Sending Reply—Blocking Consultation

Agent Unsuccessfully Consults to Another Agent Before Sending Reply

Figure 78 on page 143 shows an agent's unsuccessful attempt to consult with another agent before sending a reply.

The interaction is routed to Agent 1. Agent 1 accepts the inbound e-mail (IRF1) and creates an outbound reply, closing the original inbound e-mail (IRF2). Agent 1 initiates a Consult interaction (IRF3) and transfers it to Agent2. Agent 2 does not accept the invitation into the Consult interaction (IRF4). Agent 1 continues working on the outbound reply (IRF2). The outbound reply is placed into Outbound Queue, from which Outbound Strategy sends it out of the contact center to the customer.

Note: The "Talk" box in the Consult portion of the diagram is not shaded, because the customer is not considered to be present during offline consultations.

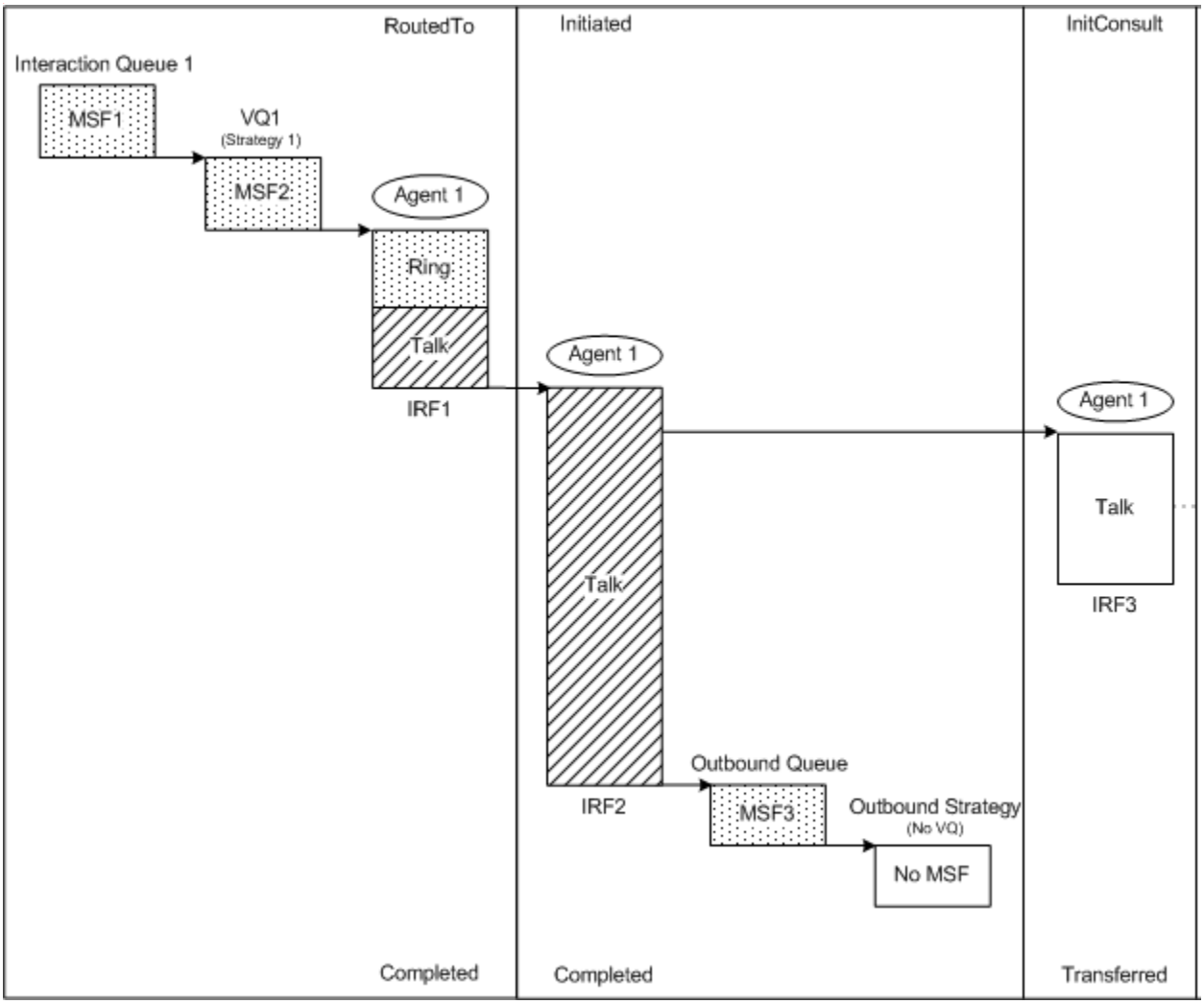

Figure 78: Agent Unsuccessfully Consults to Another Agent Before Sending Reply

Agent Saves Draft Reply Before Sending (Mediation)

Figure 79 on page 144 shows the outcome of an e-mail interaction that is routed to an agent who replies to the e-mail, after first saving an initial version of the reply in a workbin, where the time in the workbin is considered to be mediation.

The interaction is routed to Agent 1. Agent 1accepts the inbound e-mail (IRF1) and creates an outbound reply, closing the original inbound e-mail (IRF2). Agent 1 saves the outbound reply e-mail in a Draft Workbin. Later, Agent 1 pulls the reply e-mail from the workbin, makes some final modifications to the reply, and then places the reply into Outbound Queue, from which Outbound Strategy sends it out of the contact center to the customer (IRF3).

For the outcome of the same interaction flow when the workbin time is considered to be hold, see Figure 81 on page 146.

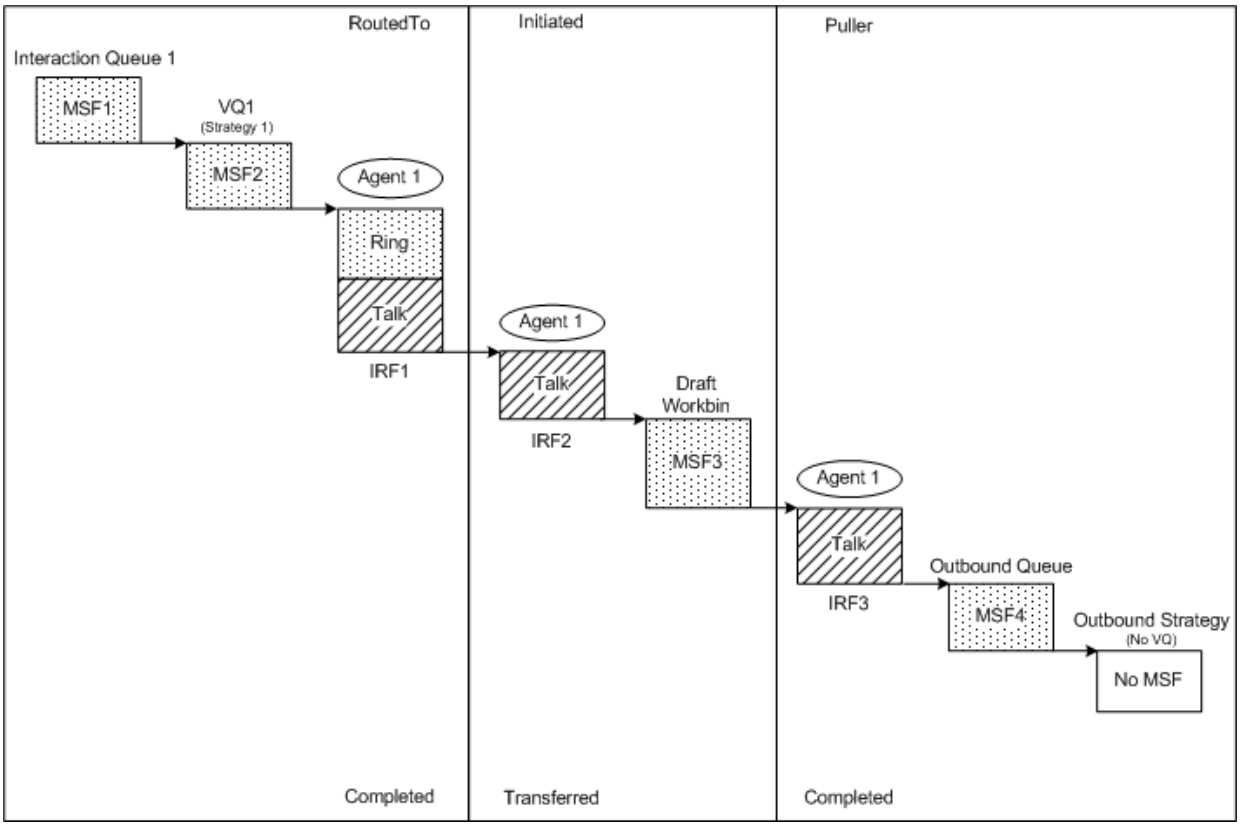

Figure 79: Agent Saves Draft Reply Before Sending (Mediation)

Agent Pulls E-Mail from an Interaction Queue or Workbin

Figure 80 shows the outcome when an e-mail interaction is pulled from an Interaction Queue or Workbin (MSF1) for further handling by Agent 1. After working on the e-mail, Agent 1 places the interaction into another Interaction Queue or Workbin (MSF2), for further processing.

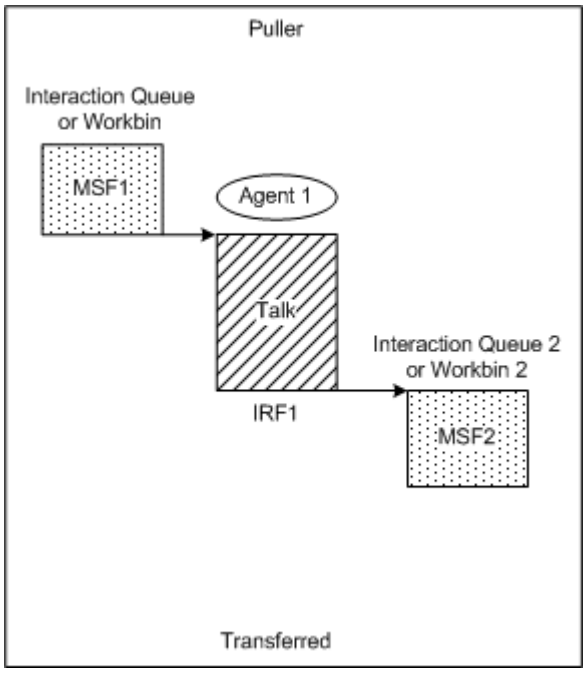

Figure 80: Agent Pulls E-Mail from Workbin

Agent Saves Draft Reply Before Sending (Hold)

Figure 81 on page 146 shows the outcome of an e-mail interaction that is routed to an agent who replies to the e-mail, after first saving an initial version of the reply in a workbin, where the time in the workbin is considered to be hold, instead of mediation. For information about when workbin time is considered to be hold, see the description of the `populate-workbin-as-hold` option in the *Genesys Info Mart 8.1 Deployment Guide*.

The interaction is routed to Agent 1. Agent 1accepts the inbound e-mail (IRF1) and creates an outbound reply, closing the original inbound e-mail (IRF2). Agent 1 saves the outbound reply e-mail in a Draft Workbin. Later, Agent 1 pulls the reply e-mail from the workbin, makes some final modifications to the reply, and then places the reply into Outbound Queue, from which Outbound Strategy sends it out of the contact center to the customer.

For the outcome of the same interaction flow when the workbin time is considered to mediation, see Figure 79 on page 144.

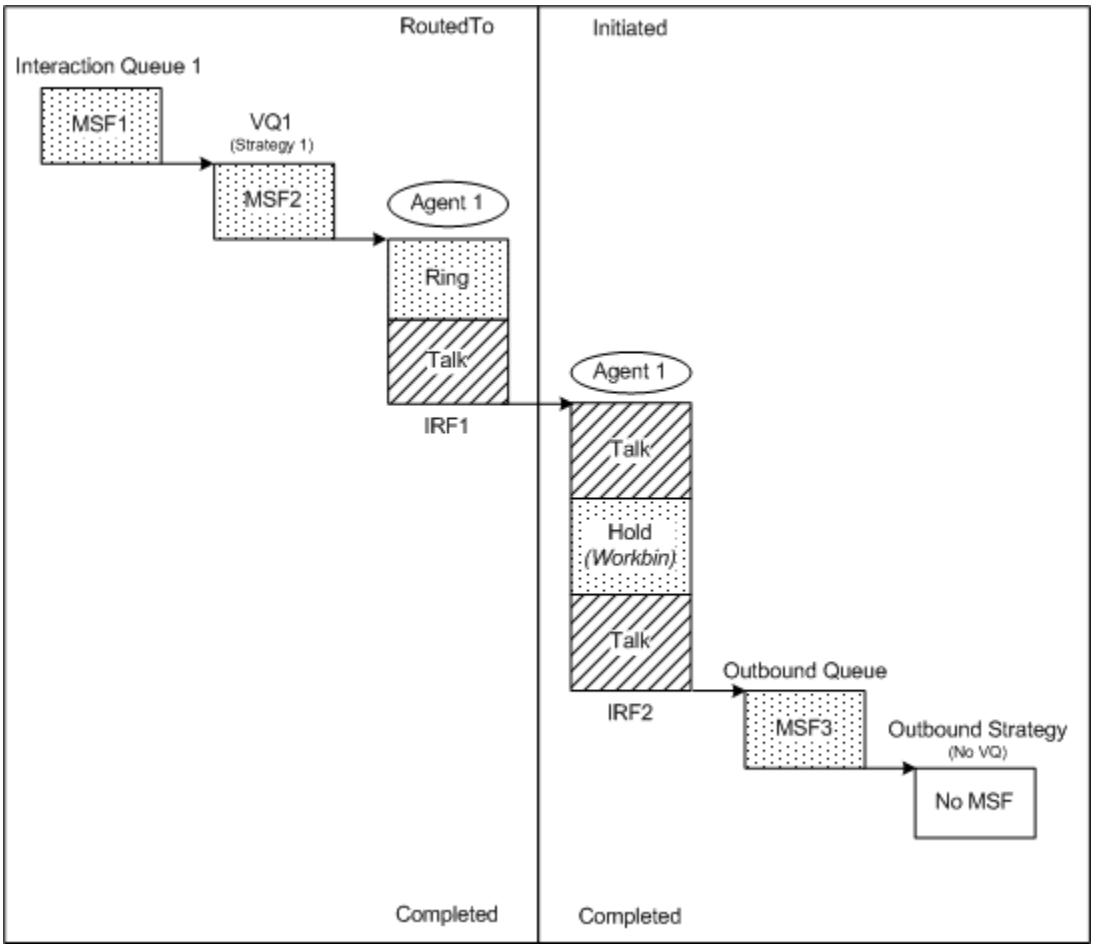

Figure 81: Agent Saves Draft Reply Before Sending (Hold)

Multipart Reply

Figure 82 on page 147 shows the outcome of a multipart reply.

In this example, Agent 1 initiates two outbound replies. For example, Agent 1 may have received a new customer order (IRF1) and initiated one reply, which is to be completed by Agent 2 in Shipping (IRF2). At the same time, Agent 1 also initiated a reply to the customer, confirming the order and providing billing-related information. Agent 1 transfers the first outbound reply directly to Agent 2 (IRF3). Agent 1 remains in this outbound reply until Agent 2 accepts the invitation into the interaction. Agent 2 later completes this reply, placing it in an Outbound Queue, from which Outbound Strategy sends it out of the contact center. Agent 1also creates and completes the second outbound reply (IRF4), now closing the original inbound interaction, and places this second reply in an Outbound Queue, from which Outbound Strategy sends it out of the contact center. In this example, the second reply that Agent 1 created was actually the first reply sent to the customer.

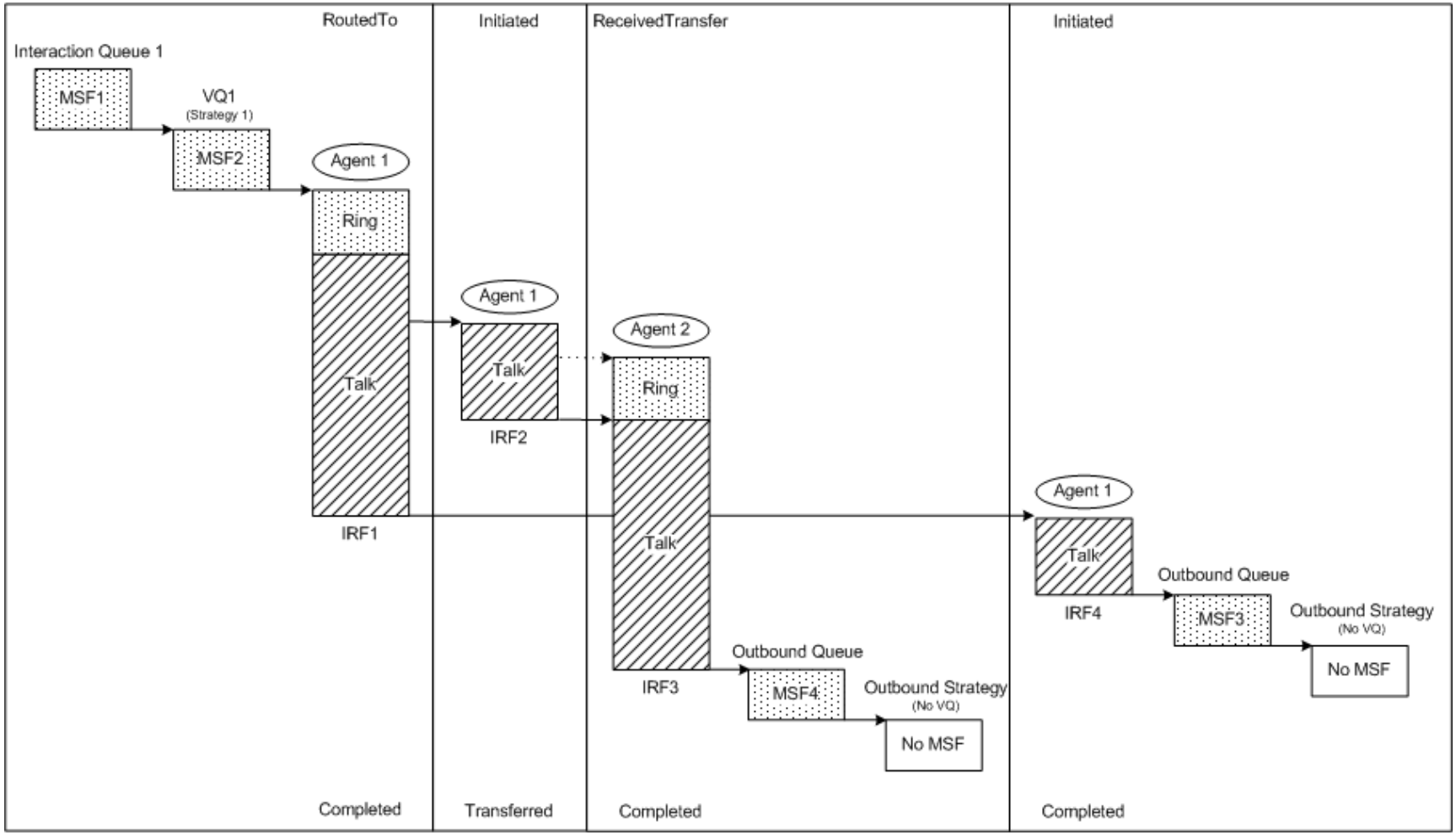

Figure 82: Agent Sends Multipart Reply

Multipart E-Mail Reply with Unsent Reply

Figure 83 on page 149 shows the outcome of a multipart reply in which one of the replies was stopped without being sent.

This example illustrates the `OutboundStopped` technical result. The `OutboundStopped` technical result applies to any outbound multimedia interaction that is stopped without being sent; it is not limited to outbound replies. In this example, the outbound reply initiated by Agent 1 (IRF2), and transferred to Agent 2, is stopped by Agent 2 (IRF3). Agent 2 worked on the outbound reply for some time, but then stopped the reply without sending it. In the meantime, as in Figure 82 on page 147, Agent 1 worked on a second reply, which was sent to the customer (IRF4)

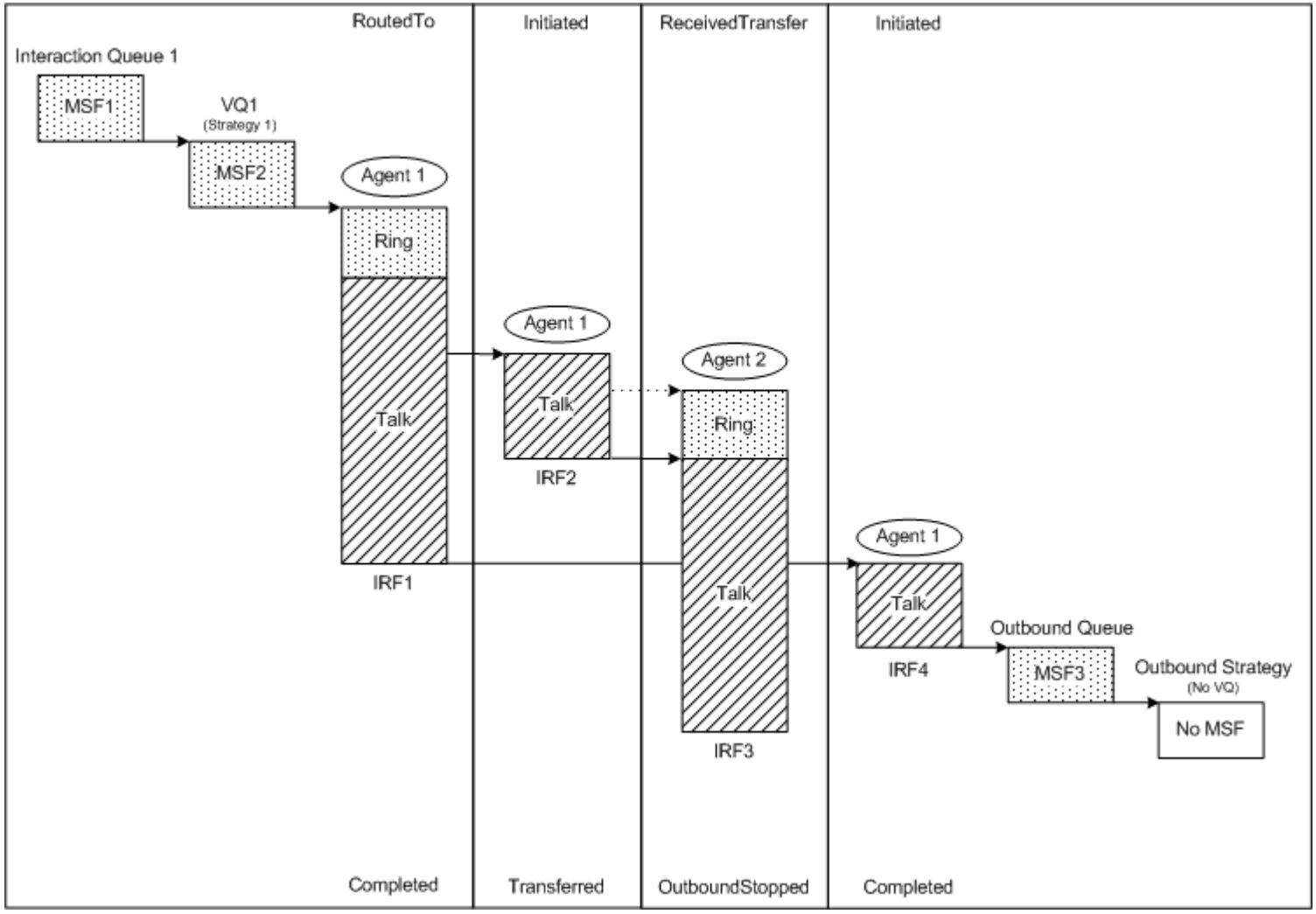

Figure 83: Multipart E-Mail Reply with Unsent Reply

Routing Strategy Repeatedly Fails to Find a Target

Figure 84 shows the outcome of an inbound e-mail interaction that is eventually routed to an agent after the routing strategy repeatedly failed to find an available agent.

In this example, Strategy 1 puts the interaction back into Interaction Queue 1 several times, until the strategy finally routes the interaction successfully to Agent 1 (IRF1). All the unsuccessful attempts to select an agent are combined into one mediation (MSF1 for Interaction Queue 1), except for the last attempt, which is when the interaction was successfully routed (MSF2 for VQ1). Given the focus of this example, the diagram does not show the continued processing after Agent 1 gets control of the interaction; the technical result for IRF1 depends on what Agent 1 does with the interaction.

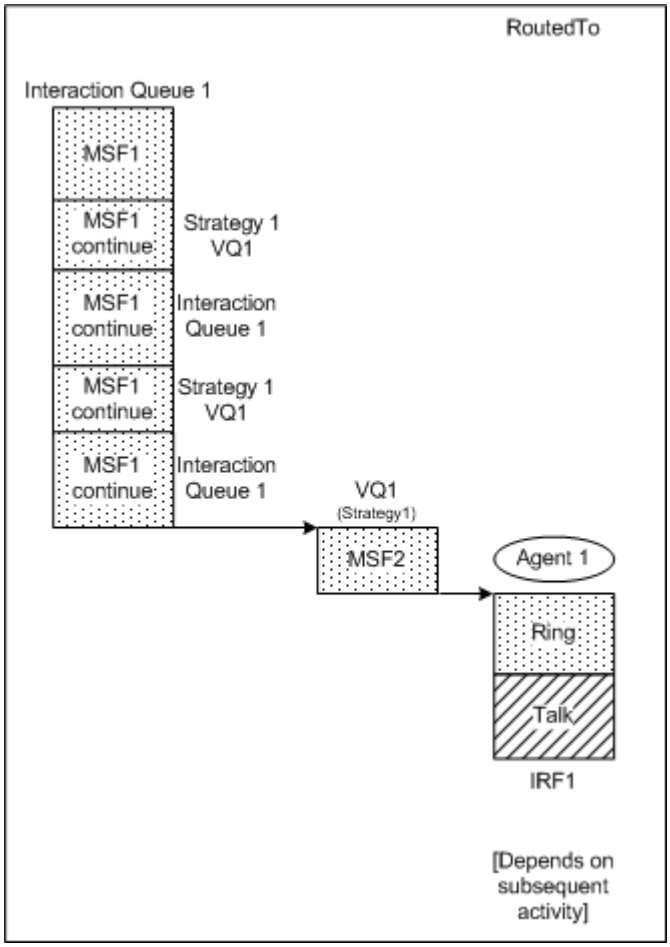

Figure 84: Strategy Routes Interaction After Repeatedly Unsuccessful Routing Attempts

Chat Interactions

This section contains several examples of chat flows. Each example represents a different outcome:

- A routing strategy routes the chat interaction to an agent, and the agent replies (see page 151).
- A routing strategy routes the chat interaction to an agent, but the agent does not accept the invitation (see page 152).
- A routing strategy routes the chat interaction to an agent, who transfers it to another agent (see page 153).
- A routing strategy routes the chat interaction to an agent, who unsuccessfully attempts to transfer it to another agent (see page 154).
- A routing strategy routes the chat interaction to an agent, who conferences in another agent (see page 155).
- An agent attempts to conference another agent in to chat with the customer, but fails (see page 156).
- The customer abandons the chat interaction while waiting in the interaction queue (see page 157).
- The customer abandons the chat interaction during routing (see page 158).
- A routing strategy routes the chat interaction to an agent, but the customer abandons the interaction while the agent was being alerted (see page 158).

Strategy Delivers Chat to Agent, and Agent Replies

Figure 85 on page 152 shows the outcome of a chat interaction that a routing strategy routes to an agent, who accepts the invitation and chats with the customer.

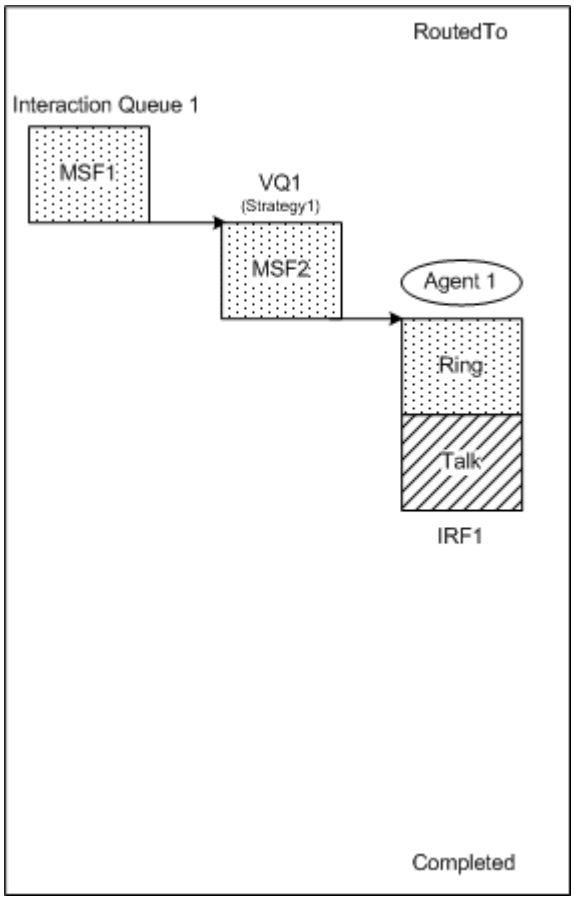

Figure 85: Strategy Routes Chat to Agent, and Agent Replies

Agent Invited into Chat and Invitation Revoked

Figure 86 on page 153 shows the outcome of a chat interaction that a routing strategy routes to an agent, who does not accept the invitation. The chat interaction is returned to the interaction queue so that other routing targets can be evaluated.

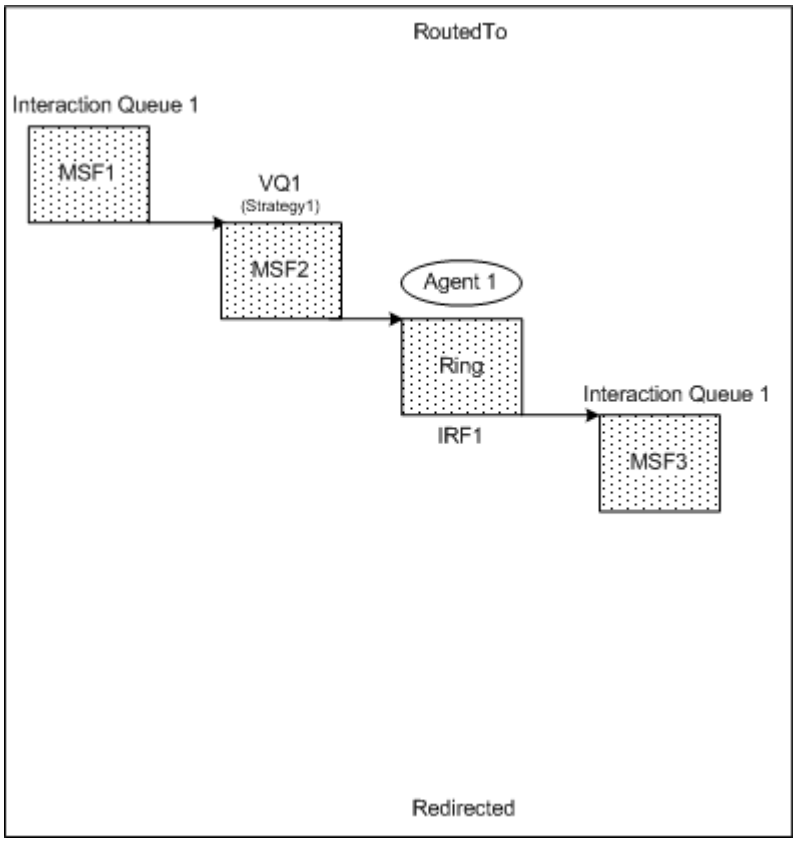

Figure 86: Agent Does Not Accept Chat Invitation

Agent Transfers Chat to Another Agent

Figure 87 on page 154 shows the outcome of a chat interaction that is routed to an agent, who transfers the chat to another agent.

The interaction is routed to Agent 1. Agent 1 accepts the invitation into the interaction and chats with the customer (IRF1). Agent 1 then transfers the chat interaction to Agent 2, who accepts the invitation and then chats with the customer (IRF2).

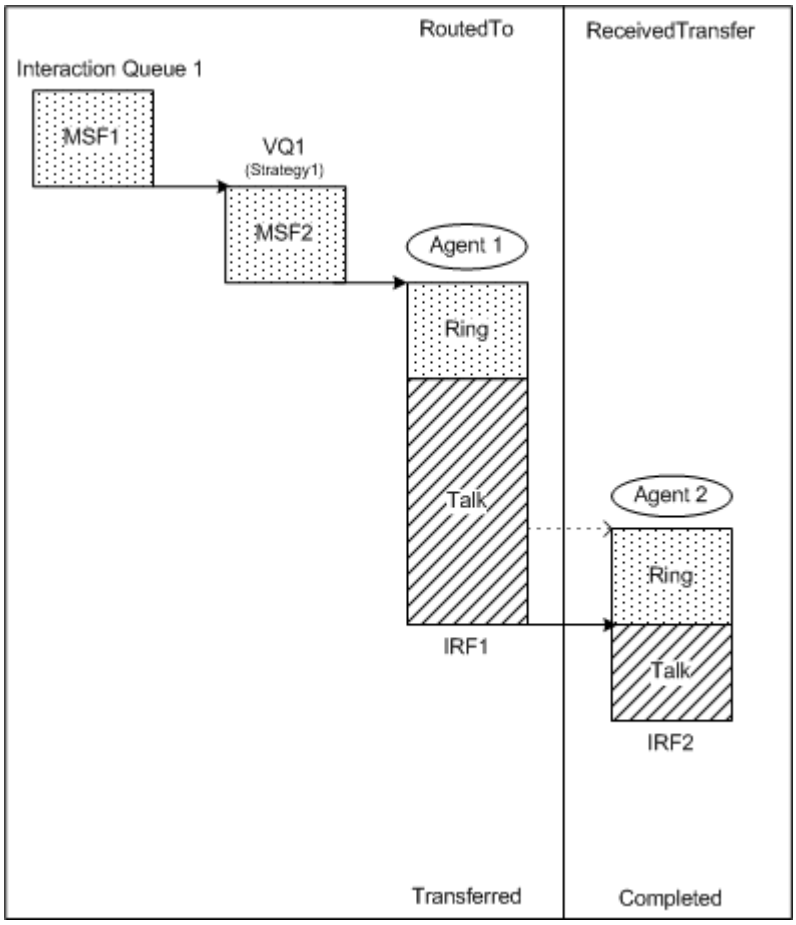

Figure 87: Agent Transfers Chat to Another Agent

Agent's Attempt to Transfer Chat to Another Agent Fails

Figure 88 on page 155 shows the outcome of an unsuccessful attempt to transfer a chat interaction to another agent.

The interaction is routed to Agent 1. Agent 1 accepts the invitation into the interaction and chats with the customer (IRF1). Agent 1 then attempts to transfer the chat interaction to Agent 2. Agent 2 does not accept the invitation (IRF2). Agent 1 remains in the chat interaction (IRF1), and in this example, completes the chat with the customer.

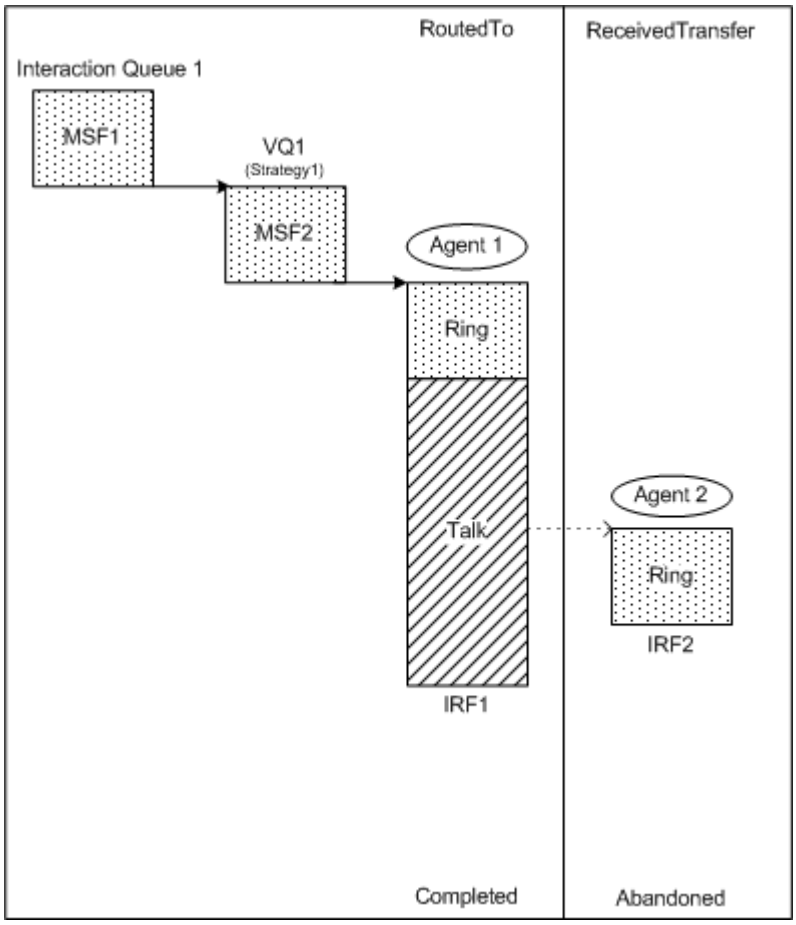

Figure 88: Unsuccessful Agent-to-Agent Chat Transfer Attempt

Agent Conferences In Another Agent

Figure 89 on page 156 shows the outcome of a chat interaction that is routed to an agent, who conferences in another agent.

The interaction is routed to Agent 1. Agent 1 accepts the invitation into the interaction and chats with the customer (IRF1). Agent 1 then attempts to conference in Agent 2. Agent 2 accepts the invitation and then also chats with the customer (IRF2).

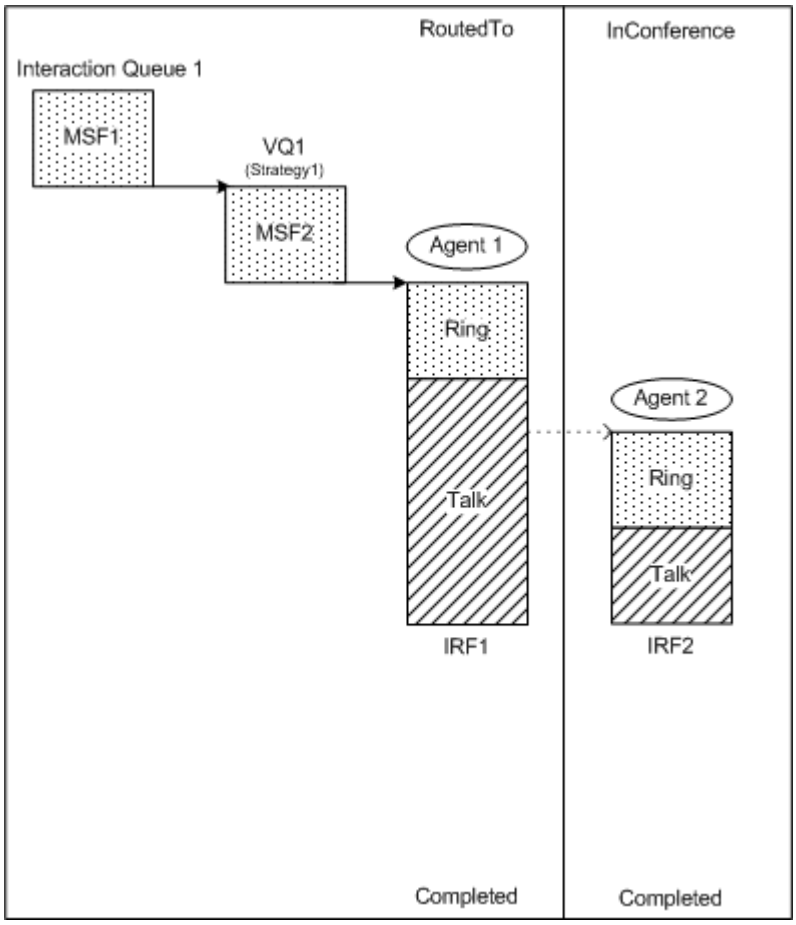

Figure 89: Agent Conferences in Another Agent

Agent's Attempt to Conference in Another Agent Fails

Figure 90 on page 157 shows the outcome of an unsuccessful attempt to conference another agent into a chat interaction.

The interaction is routed to Agent 1. Agent 1 accepts the invitation into the interaction and chats with the customer (IRF1). Agent 1 then attempts to conference in Agent 2. Agent 2 does not accept the invitation (IRF2). Agent 1 remains in the chat interaction (IRF1), and in this example, completes the chat with the customer.

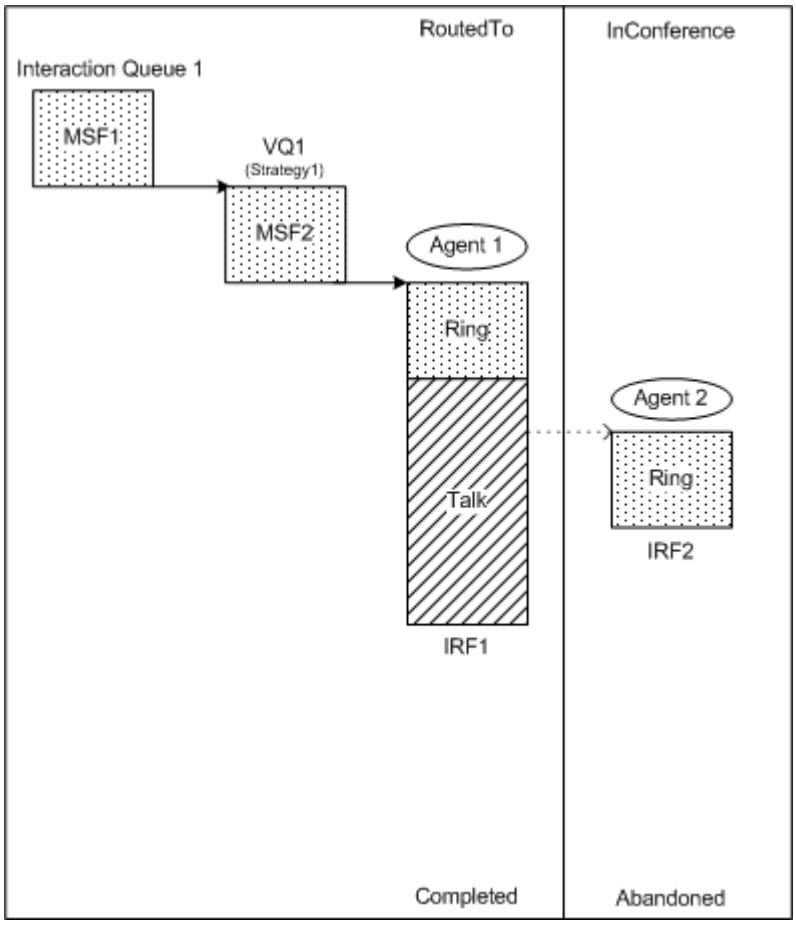

Figure 90: Agent Attempts to Conference in Another Agent. but Fails

Customer Abandons Chat in Queue

Figure 91 shows the outcome of a chat interaction that is submitted to an inbound interaction queue, but is abandoned by the customer while it is in the interaction queue.

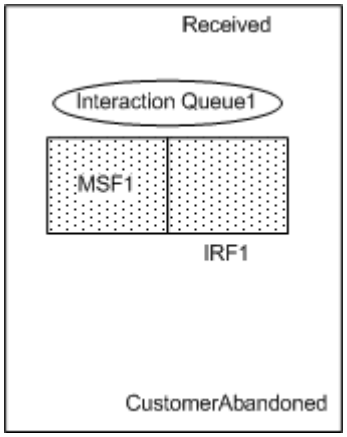

Figure 91: Customer Abandons Chat in Queue

Customer Abandons Chat During Routing

Figure 92 shows the outcome of a chat interaction that is submitted to an inbound interaction queue, but is abandoned by the customer while a routing strategy is attempting to route the interaction.

The resource role of `Puller` is used because the strategy pulled the interaction from the interaction queue.

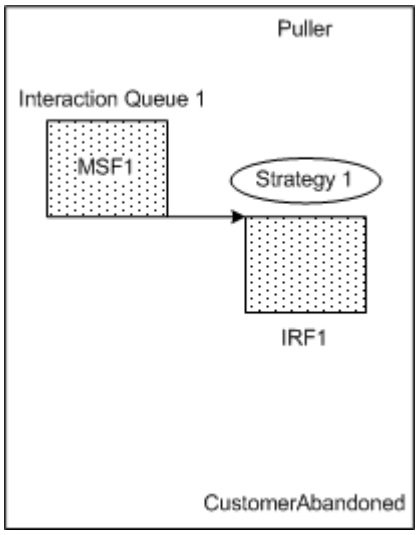

Figure 92: Customer Abandons Chat During Routing

Customer Abandons Chat During Agent Alerting

Figure 93 on page 159 shows the outcome of a chat interaction that is abandoned while the agent is being alerted.

The interaction is routed to Agent 1. The customer abandons the chat interaction before the agent accepts the invitation.

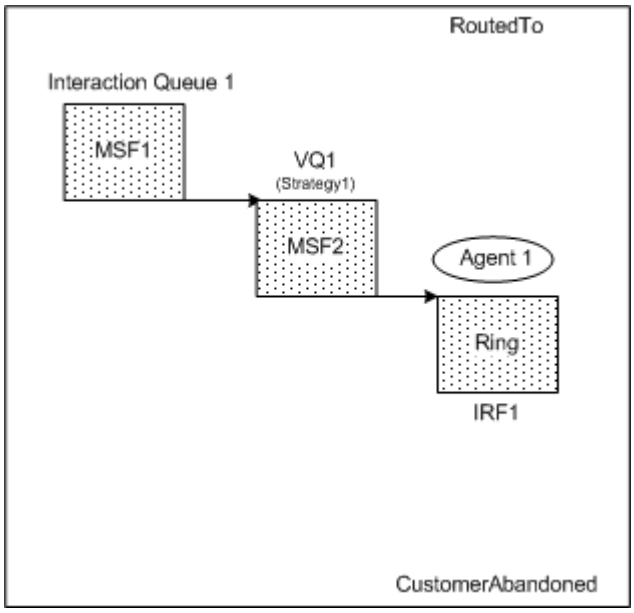

Figure 93: Customer Abandons Chat During Agent Alerting

Chapter

6 Data Lineage: Voice of Data

This chapter describes how Genesys Info Mart tracks data, enabling you to understand what data is collected and how it is processed. This information can be used for reporting and to assess data accuracy (data validation). This chapter contains the following sections:

Data Lineage Overview

Data lineage provides information that records history of job execution and data transform for each piece of data. Data stored as part of data lineage allows for bi-directional data tracking and enables you to answer the following questions:

- What process created the piece of data?
- What was the source of the data?
- What data was created by a specified job?
- What data in the system was created based on specified source data?

Data lineage has two aspects:

- **Voice of data**—This feature pertains to data-quality validation and troubleshooting. It enables you to trace a particular data item in a source system based on data in the target system, and also to trace data in the opposite direction (from source to target).
- **Voice of process**—Provides data processing history, and traces which ETL process created this piece of data. It also traces in the opposite direction (from process to data).

Note: This chapter focuses on voice of data. For information about voice of process, see the *Genesys Info Mart 8.1 Operations Guide*. For detailed descriptions of the tables and views related to Data Lineage, see the *Genesys Info Mart8.1 Reference Manual* for your RBDMS.

Voice of Data

Voice of data functionality enables you to trace data origins or targets.

To use voice of data, you must be familiar with which GIDB tables provide which kind of data. This information, called *static mapping*, cannot be derived from the schema. The connections are presented in Table 8 and discussed in more detail below.

Table 8: Data Mapping

Genesys Info Mart		GIDB	
Table	**Field**	**Table**	**Field**
INTERACTION_FACT	MEDIA_SERVER_IXN_ GUID	GIDB_G_CALL_V, GIDB_G_CALL_MM	CALLID
INTERACTION_ RESOURCE_FACT	INTERACTION_RESOURCE _ID[a]	GIDB_G_PARTY_V, GIDB_G_PARTY_MM	PARTY_KEY
INTERACTION_ RESOURCE_FACT	PARTYGUID	GIDB_G_PARTY_V, GIDB_G_PARTY_MM	PARTYGUID
MEDIATION_ SEGMENT_FACT	MEDIA_SERVER_IXN_GUID	GIDB_G_CALL_V, GIDB_G_CALL_MM	CALLID
	MEDIATION_GUID	GIDB_G_PARTY_V, GIDB_G_PARTY_MM GIDB_VIRTUAL_QUE UE_V, GIDB_VIRTUAL_QUE UE_MM	PARTYGUID VQID
RESOURCE_	RESOURCE_CFG_DBID, RESOURCE_CFG_TYPE_ID	GIDB_GC_AGENT, GIDB_GC_ENDPOINT, GIDB_GC_SCRIPT	ID (dbid)
GROUP_	GROUP_CFG_DBID	GIDB_GC_GROUP	ID

Table 8: Data Mapping (Continued)

Genesys Info Mart		GIDB	
Table	**Field**	**Table**	**Field**
PLACE	PLACE_CFG_DBID	GIDB_GC_PLACE	ID
SKILL	SKILL_CFG_DBID	GIDB_GC_SKILL	ID

a. The PARTY_KEY of the last party in the Interaction Resource Fact (IRF) record is used as the INTERACTION_RESOURCE_ID. Primary keys can match in many-to-one relationship.

The information in this table is a starting point that enables you to:

- Trace a record back to the source records that triggered its creation, or trace data from its origin to its final target.
- Select an interaction in Genesys Info Mart and pull out from ICON database all corresponding records.
- Trace the information in these records back to the source application logs (for example, the T-Server, SIP Server, or Interaction Server logs).

How to Use Voice of Data

The basic concept of Voice of Data is that each interaction can be traced from its initial entry into your environment through its conclusion by way of specific linking IDs that enable you to trace, in either direction, the processing records from IDB to the Genesys Info Mart database via the GIDB.

To accomplish this, each of the interaction records in the Info Mart fact tables are traceable back to source table in GIDB using keys that indicate for a particular record from which records in the source tables it was created. Multiple links enable you to trace the interaction records in the fact tables to the GIDB, from where we can find records in ICON databases and application logs.

For example, the INTERACTION_RESOURCE_FACT (IRF) table stores the PARTYGUID of the handling party. A simple join between GIDB_G_PARTY with either G_PARTY or PARTYGUID links the IRF table with the corresponding party record in GIDB/IDB.

Use Cases

The examples in this section show some specific ways to use Voice of Data. These provide only a small sampling of the sorts of questions you can answer using Voice of Data.

Identify Source of GIDB Data

"How do I know where specific GIDB data comes from?"

To identify the source for GIDB data, join the GIDB table of interest with the `CTL_DS` table using `DATA_SOURCE_KEY, CTL_DS`. The value for `DS_DBID` in the resulting data set provides the value of the data source DBID (for example DBID of the T-Server for voice call data).

Identify the Source ICON Instance

"We use multiple instances of ICON. Which one created this specific record?"

To identify instance of ICON which was the source of a particular record in GIDB, use the `GSYS_SYS_ID` field in the GIDB record. This field contains the DBID of the ICON application as defined in the Configuration Layer.

Chapter

7 Representing Dates and Times of Day

This chapter describes how Genesys Info Mart represents dates and times of day. Because of the large volume of data handled by Genesys Info Mart, most SQL queries of a fact table are constrained by date and time.

This chapter contains the following sections:

Dates and Times of Day

Dates and times of day are stored in the `START_TS` and `END_TS` fields, which mark the start and end of each handling stage. The `START_DATE_TIME_KEY` and `END_DATE_TIME_KEY` reference the `DATE_TIME` dimension, which exists in all fact tables.

Dates and times are stored in Coordinated Universal Time (UTC) format. You can express local, enterprise, or tenant time using custom `DATE_TIME` dimensions that offset the UTC time by a specified amount of time.

Note: For instructions on configuring custom `DATE_TIME` tables, see the *Genesys Info Mart 8.1 Deployment Guide*.

Since in UTC, `0` = `1970 January 1`, each custom `DATE_TIME` dimension table will associate this UTC key with a different local time that is relevant for your enterprise. This enables you to use the same keys to create reports in different time zones.

How Dates and Times Can Be Constrained

Each fact table row has a surrogate key, START_DATE_TIME_KEY, that references the DATE_TIME dimension that represents its start date and time. This surrogate key can constrain the fact table rows by start date and time of day. Similarly, the END_DATE_TIME_KEY can be used to constrain the fact table rows by end date and time of day.

Each fact table row contains measurements that represent the start date and time of day, and the end date and time of day. These measurements can constrain fact table rows by any arbitrary time span, based on whether the fact table row:

- Starts and ends within the time span.
- Starts before, and ends within, the time span.
- Starts within, and ends after, the time span.
- Starts before, and ends after, the time span.

In any case, you must create the appropriate database indexes in order to efficiently retrieve the data you want.

All fact tables have surrogate key references to the DATE_TIME dimension that represent the 15-minute date and time interval in which a fact started and ended.

The DATE_TIME dimension is useful for constraining based on an arbitrary range of 15-minute time intervals, because this single dimension includes both date and time of day. The dimension keys increase regularly each 15 minutes.

Working with Timestamps and the DATE_TIME Dimension

The following example illustrates how Genesys Info Mart represents the date and time of an inbound call in local time.

Example

An inbound call arrives at a contact center in San Francisco on October 21, 2009 at 5:05 PM local time (PDT). This time corresponds to 1:05 AM on October 22, 2009 in UTC GMT time zone, or 1256173500 seconds, expressed in UTC integer format. This integer is stored in the START_TS field in the table containing data about the call.

The call's start time also falls into a 15-minute time interval that begins on October 22, 2009 at 1:00 AM in the UTC GMT time zone, or 1256173200 seconds in UTC integer format. This integer is stored in the START_DATE_TIME_KEY field in the tables containing data related to the call.This value is a surrogate key that can be used to link to the corresponding

DATE_TIME_KEY field in any DATE_TIME_CUSTOM dimension. These custom tables contain text labels for the day of the week, month, year, and so on, in whichever local time zone formats your business requires.

In this example, a DATE_TIME_CUSTOM table has been created for the Pacific time zone containing labels in local PDT format. The START_DATE_TIME_KEY field in the fact table containing the UTC integer 1256173200 (corresponding to 5:00 pm PDT), can be used to link to this DATE_TIME_CUSTOM dimension. The correct text labels for the Pacific time zone can then be retrieved for your reports.

Calculating Timestamps

To show timestamps in reports converted to a particular time zone, use a simple calculation combining the START_TS (or END_TS) field of a fact table with the DATE_TIME_KEY and CAL_DATE fields of the DATE_TIME_CUSTOM table created for that time zone.

For example, to convert the timestamp value, 1256173500, from the Example on page 166, where the time of call arrival is stored in UTC seconds format in the START_TS field of the corresponding INTERACTION_RESOURCE_FACT (IRF) row in a Microsoft SQL Server RDBMS, execute the following query on the custom DATE_TIME_CUSTOM dimension and IRF table:

```
select DTC.CAL_DATE + CAST ((IRF.START_TS - DTC.DATE_TIME_KEY) as
float) / CAST (86400 as float)
from DATE_TIME_CUSTOM DTC, INTERACTION_RESOURCE_FACT IRF
where DTC.DATE_TIME_KEY = IRF.START_DATE_TIME_KEY
```

The resulting value is October 21, 2009 at 5:05 pm in PDT time zone.

To make the same conversion in an Oracle RDBMS, execute the following query:

```
select DTC.CAL_DATE + (IRF.START_TS - DTC.DATE_TIME_KEY) / 86400
from DATE_TIME_CUSTOM DTC, INTERACTION_RESOURCE_FACT IRF
where DTC.DATE_TIME_KEY = IRF.START_DATE_TIME_KEY
```

Calendar Years and Week-Numbering Years

There are two available ways to number the weeks in a year:

- **Full-week numbering**—In this system, weeks always contain seven days and always start on the day of the week specified as Day 1 in the first-day-of-week configuration option. This system supports the ISO-8601 week configuration used in the European Union and Russia.

- **Simple-week numbering**—This results in the week calendar matching the calendar year. Week 1 begins on January 1. As a result, the first day of the week differs each year. Most of the time, Weeks 1 and 52 will have fewer than seven days. This is the functionality used in previous releases of Genesys Info Mart.

Week-Numbering Table Fields

The DATE_TIME table contains several fields that are used to support the full-week numbering system.

- WEEK_YEAR—This column stores a Week Numbering Year. This year may be different from Calendar year.

  For example, in ISO-8601, 31 December of 2007 is Week 1 Day 1 of 2008. So in this case we have 2007 as the Calendar year and 2008 as the Week Numbering year.
- LABEL_YYYY_WE_D—The label for the day of the week.
- LABEL_TZ—This field stores the time zone offset.

Week-Numbering Configuration Options

A number of configuration options control how week-numbering is done. For more information, see the descriptions of the [date-time] options in the configuration options reference chapter in the *Genesys Info Mart 8.1 Deployment Guide*.

Note: In deployments that include Reporting and Analytics Aggregates (RAA) or custom aggregation that uses the DATE_TIME table, Genesys strongly recommends that you do not change [date-time] options that set basic features of the calendar—such as the time zone—after the aggregation engine has started to aggregate data. If you do, at the very least, you will likely need to re-aggregate data.

Maintaining the Calendar Dimensions

The DATE_TIME dimension is a default calendar that is set up when Genesys Info Mart is initialized and that needs to be populated ahead of time on a continuing basis. Maintenance of the DATE_TIME dimension is controlled by the date-time-min-days-ahead and date-time-max-days-ahead configuration options in the [date-time] configuration section. Similarly, maintenance of any custom calendar dimension(s) is controlled by equivalent options in the applicable [date-time-*] configuration section(s).

Job_MaintainGIM adds records to the calendar table if the last existing record is earlier than <current-date> + date-time-min-days-ahead. Records are added until <current-date> + date-time-max-days-ahead.

For example, take a scenario in which date-time-min-days-ahead is set to 183, date-time-max-days-ahead is set to 366, and today is March 30, 2011. In other words, <current-date> + date-time-min-days-ahead = September 29, 2011.

Case 1: `DATE_TIME` is populated until January 1, 2012.

Since (March 30, 2011 + 183) < January 1, 2012, `Job_MaintainGIM` will not add any records to the calendar table.

Case 2: `DATE_TIME` is populated until June 1, 2011.

Since (March 30, 2011 + 183) > June 1, 2011, `Job_MaintainGIM` will add records to the calendar table until (March 30, 2011 + 366) = March 30, 2012.

Chapter

8 Data-Quality Considerations

This chapter discusses the implications for data quality of certain challenging aspects of extract, transform, and load (ETL) processing. It contains the following sections:

Partially Merged Calls

In multi-site scenarios, Genesys Info Mart must merge data from the multiple sites. If an interaction moves from site to site during the handling process, Genesys Info Mart uses linkage data to integrate the data from various T-Servers into a single interaction. This can be disrupted for a number of reasons, causing data-quality issues.

There are three major cases when data from a site might be unavailable:

- The site is not monitored (partially monitored environment).
- The site is monitored, but information is missing.
- Information is delayed.

Data Issues in a Partially Monitored Environment

If you configure Genesys Info Mart to extract voice interaction data from topologies in which not all T-Servers or IVR Servers involved in the call flow

are monitored by Interaction Concentrator, data inconsistencies can occur, such as incomplete and missing data.

Note: If you have an environment that includes unmonitored sites, such sites must be noted in the GSYS_DNPREMOTELOCATION table.

A partially monitored environment can result in missing data at the start, middle, or end of an interaction. The following interaction scenarios can affect the population of interaction data within Genesys Info Mart, resulting in data inconsistencies:

- The interaction originates in an unmonitored T-Server.
- The interaction terminates in an unmonitored T-Server.
- The interaction originates in a monitored T-Server, moves to an unmonitored T-Server, and then passes on to a monitored T-Server.

Note: Any combination of the previously mentioned deployments will have an effect on the population of voice interaction data.

Each time that the interaction moves from an unmonitored to a monitored T-Server, it appears to be a new interaction. For example, a single interaction might start on a monitored T-Server, be sent to an unmonitored T-Server, and then be sent to a monitored T-Server. This single interaction is represented in Genesys Info Mart as two interactions. When this type of interaction scenario occurs, the linkage information that ties an interaction together as it moves from T-Server to T-Server is incomplete and Interaction Concentrator cannot associate what it sees as multiple calls into a single interaction.

A partially monitored deployment can result in data that is incorrect or missing from the following fact tables:

- Interaction Fact—Some interaction facts will be missing where entire calls or parties are missing in the source data. Tables that are affected by this can include:
  - Interaction Resource Fact
  - Ixn Resource State Fact
  - Mediation Segment Fact
- Interaction Resource Fact (IRF)—Some IRFs might not reflect accurate information about MEDIATION RESOURCES, CONSULT, CONFERENCE, and TRANSFER metrics, TECHNICAL DESCRIPTOR, ROUTING TARGET and SERVICE LEVEL flags, such as MET SERVICE OBJECTIVE and SHORT ABANDONED. This occurs because these fields are highly dependent on other resources that are involved in the interaction which might or might not be monitored.

- Ixn Resource State Fact (IRSF)—Some IRSFs might not contain accurate information in the `STATE DESCRIPTOR` and `ROLE VALUE` for the states that are generated for IRFs. This occurs because these values are populated based on interaction-type information that might change as a result of an unmonitored T-Server in the environment.
- Mediation Segment Fact (MSF)—Some MSFs might be missing or have incorrect `TECHNICAL DESCRIPTOR` values because the ETL cannot determine why the interaction was placed in the Queue or virtual queue, or whether it was answered or abandoned after it was distributed from the queue or virtual queue.

Late Data

Late data from sources that Genesys Info Mart considers to be currently active can be a result of various issues, such as intermittent connectivity issues for an IDB. For example, you might currently have data for only the beginning of an interaction, but data from a second T-Server is anticipated to arrive "soon". In this case, *soon* means that data should arrive before the timeout set in the Genesys Info Mart `extract-data-stuck-threshold` configuration option, which specifies the allowable delay to wait for the missing data to become available, or before the merge timeout that is controlled by the `max-call-duration` option. If the delayed data arrives before the stuck threshold timeout expires, the interaction is processed normally. If the threshold expires before the data arrives, the data is treated as missing.

For more information about the `extract-data-stuck-threshold` timeout, see the configuration options reference chapter in the *Genesys Info Mart 8.1 Deployment Guide*. For the definition of what Genesys Info Mart considers to be *active data sources,* see the section about terminology in the Overview chapter in the *Genesys Info Mart 8.1 Deployment Guide*.

Error Handling in Case of Missing Data

For various reasons, information from a data source for a specific time range might be missing from a monitored site. In an HA environment, a single failure does not result in loss of data. If, in exceptional circumstances, multiple failures occur, the HA environment becomes, in effect, a non-HA environment.

For a full discussion of the potential points of failure in a non-HA environment and the implications for data quality, see the section on error handling in the chapter about ETL processing in the *Genesys Info Mart 8.1 Deployment Guide*.

Configuration Options for Missing-Data Behavior

For transformation, two configuration options control the handling of missing data: `error-policy-islink-dangling` and `error-policy-irf-exception`. The

first option enables you to determine whether missing data is handled as an exception. If it is, the second option enables you to specify how such an exception is handled.

For more information about Genesys Info Mart error-handling, see the chapter about ETL processing in the *Genesys Info Mart 8.1 Deployment Guide*.

Missing Configuration Objects

Genesys Info Mart checks the list of known configuration objects during data transformation. If, during transformation of configuration facts data, `Job_TransformGIM` notes a missing configuration object, it records the information in the `STG_IDB_FK_VIOLATION` table.

During transformation of data types other than configuration data, Genesys Info Mart treats such missing configuration objects as late-arriving and creates placeholders for the missing objects based on the configuration object type and its unique ID. When the missing configuration objects arrive from Configuration Server, these placeholders are populated with missing data.

Note: The unique configuration ID prevents accidental duplication of configuration objects.

High Availability

Deploying a high availability (HA) configuration can greatly reduce the possibility of data loss and other issues with data quality. Genesys recommends using HA throughout the data chain, from data sources through Interaction Concentrator. Genesys Info Mart is designed to take advantage of HA configurations in order to determine and draw on the most complete and reliable data available.

In an HA configuration, each HA set of Interaction Concentrator instances consists of two or more redundant ICONs, each populating its own IDB. Genesys Info Mart selects available data in such a way that it takes the most complete and correct set of data from one of the redundant IDBs with no duplications. To accomplish this, Genesys Info Mart uses data-session information that ICON stores in each IDB to identify whether the data for the time period that is to be extracted is complete and correct. Genesys Info Mart extracts data from whichever one of the redundant IDBs has the most complete and reliable data for a particular time range.

This approach to identifying the best data for any period eliminates the need for resource-consuming double-extraction, analysis, and deduplication processes.

Criteria for Choice of Better IDB

The rules that govern how the ETL process determines the time slices within an extraction cycle, as well as how it selects the best IDB source for each time

slice, are designed to minimize both data loss and the number of switchovers from one IDB to another. For more information about the criteria for choosing the best IDB source for a particular time period, see the chapter about HA in the *Genesys Info Mart 8.1 Deployment Guide*.

Note: Genesys Info Mart requires ICON to write session information to the IDBs, whether or not your environment is HA. For information about the ICON configuration settings that Genesys Info Mart requires, see the chapter about preparing Interaction Concentrator in the *Genesys Info Mart 8.1 Deployment Guide*.

Preventing Data Quality Issues when Restarting ICON

An HA configuration can eliminate multimedia data-quality issues that might arise as a result of setting the `calls-in-the-past` ICON configuration option to the required value of `true`.

If you need to restart a multimedia ICON—for example, to install an upgrade—and you do *not* have an HA configuration, information about previous parties and first values of user data keys might be missing or inaccurate. Genesys recommends that you use an HA configuration, which eliminates these issues.

For additional information about potential data-quality issues for multimedia ICONs, see the discussion about special considerations when restarting a Multimedia ICON in the chapter about managing data sources in the *Genesys Info Mart 8.1 Operations Guide*.

Genesys

Appendix

Technical Descriptor Combinations

The TECHNICAL_DESCRIPTOR dimension is a composite of the resource role, role reason, technical result, and technical result reason attributes of a particular INTERACTION_RESOURCE_FACT (IRF) or MEDIATION_SEGMENT_FACT (MSF) record.

Table 9 on page 178 summarizes the combinations of attributes that constitute the available technical descriptor dimensions, arranged in order of the TECHNICAL_DESCRIPTOR_KEY.

For more information about the meaning of the technical descriptor attributes and how they are populated for IRFs and MSFs, see Chapter 3 on page 51.

Note: The TECHNICAL_DESCRIPTOR dimension table includes some combinations of attributes that Genesys Info Mart does not use. In Table 9, a footnote in the TECHNICAL_DESCRIPTOR_KEY column indicates those technical descriptor combinations that are not used.

Table 9: Supported Technical Descriptor Combinations

TECHNICAL_DESCRIPTOR_KEY	Resource Role	Role Reason	Technical Result	Technical Result Reason
0	RECEIVED		COMPLETED	UNSPECIFIED
1	RECEIVED		ABANDONED	UNSPECIFIED
2	RECEIVED		TRANSFERRED	UNSPECIFIED
3[a]	RECEIVED		ROUTED	UNSPECIFIED
4	RECEIVED		DIVERTED	UNSPECIFIED
5	RECEIVEDTRANSFER		COMPLETED	UNSPECIFIED
6	RECEIVEDTRANSFER		ABANDONED	UNSPECIFIED
7	RECEIVEDTRANSFER		TRANSFERRED	UNSPECIFIED
8[a]	RECEIVEDTRANSFER		ROUTED	UNSPECIFIED
9	RECEIVEDTRANSFER		DIVERTED	UNSPECIFIED
10	RECEIVEDCONSULT		COMPLETED	UNSPECIFIED
11	RECEIVEDCONSULT		ABANDONED	UNSPECIFIED
12	RECEIVEDCONSULT		TRANSFERRED	UNSPECIFIED
13[a]	RECEIVEDCONSULT		ROUTED	UNSPECIFIED
14	RECEIVEDCONSULT		DIVERTED	UNSPECIFIED
15	ROUTEDTO		COMPLETED	

Table 9: Supported Technical Descriptor Combinations (Continued)

TECHNICAL_DESCRIPTOR_KEY	Resource Role	Role Reason	Technical Result	Technical Result Reason
16	ROUTEDTO		ABANDONED	
17	ROUTEDTO		TRANSFERRED	
18[a]	ROUTEDTO		ROUTED	
19	ROUTEDTO		DIVERTED	
20	DIVERTEDTO		COMPLETED	
21	DIVERTEDTO		ABANDONED	
22	DIVERTEDTO		TRANSFERRED	
23[a]	DIVERTEDTO		ROUTED	
24	DIVERTEDTO		DIVERTED	
25	INITIATEDCONSULT		COMPLETED	
26	INITIATEDCONSULT		ABANDONED	
27	INITIATEDCONSULT		TRANSFERRED	
28	INITIATEDCONSULT		CONFERENCED	
29	INCONFERENCE		COMPLETED	
30	INCONFERENCE		ABANDONED	
31	INCONFERENCE		TRANSFERRED	
32	INITIATED		COMPLETED	

Table 9: Supported Technical Descriptor Combinations (Continued)

TECHNICAL_DESCRIPTOR_KEY	Resource Role	Role Reason	Technical Result	Technical Result Reason
33	INITIATED		ABANDONED	
34	INITIATED		TRANSFERRED	
35[a]	RECEIVEDREQUEST		COMPLETED	
36	RECEIVED		CUSTOMERABANDONED	ABANDONEDWHILEQUEUED
37	RECEIVED		CUSTOMERABANDONED	ABANDONEDFROMHOLD
38	RECEIVED		CUSTOMERABANDONED	ABANDONEDWHILERINGING
39	RECEIVEDTRANSFER		CUSTOMERABANDONED	ABANDONEDWHILEQUEUED
40	RECEIVEDTRANSFER		CUSTOMERABANDONED	ABANDONEDFROMHOLD
41	RECEIVEDTRANSFER		CUSTOMERABANDONED	ABANDONEDWHILERINGING
42	ROUTEDTO		CUSTOMERABANDONED	ABANDONEDWHILEQUEUED
43	ROUTEDTO		CUSTOMERABANDONED	ABANDONEDFROMHOLD
44	ROUTEDTO		CUSTOMERABANDONED	ABANDONEDWHILERINGING
45	DIVERTEDTO		CUSTOMERABANDONED	ABANDONEDWHILEQUEUED
46	DIVERTEDTO		CUSTOMERABANDONED	ABANDONEDFROMHOLD
47	DIVERTEDTO		CUSTOMERABANDONED	ABANDONEDWHILERINGING
48	INITIATED		DESTINATIONBUSY	
49	RECEIVED		PULLED	

Table 9: Supported Technical Descriptor Combinations (Continued)

TECHNICAL_DESCRIPTOR_KEY	Resource Role	Role Reason	Technical Result	Technical Result Reason
50	RECEIVEDTRANSFER		PULLED	
51	ROUTEDTO		PULLED	
52	PULLER		COMPLETED	
53	PULLER		TRANSFERRED	
54[a]	PULLER		ROUTED	
55	PULLER		ABANDONED	
56	PULLER		ABANDONED	ABANDONEDWHILEQUEUED
57	PULLER		CUSTOMERABANDONED	
58	PULLER		CUSTOMERABANDONED	ABANDONEDWHILEQUEUED
59	PULLER		CUSTOMERABANDONED	ABANDONEDWHILERINGING
60	REDIRECTEDTO		PULLED	
61	REDIRECTEDTO		ABANDONED	
62	REDIRECTEDTO		CUSTOMERABANDONED	
63	REDIRECTEDTO		CUSTOMERABANDONED	ABANDONEDWHILEQUEUED
64	RECEIVED		CLEARED	
65	RECEIVED		CLEARED	STUCKCALL
66	UNKNOWN		NONE	

Table 9: Supported Technical Descriptor Combinations (Continued)

TECHNICAL_DESCRIPTOR_KEY	Resource Role	Role Reason	Technical Result	Technical Result Reason
67[a]	INCONFERENCE	CONFERENCEINITIATOR	COMPLETED	UNSPECIFIED
68[a]	INCONFERENCE	CONFERENCEJOINED	COMPLETED	UNSPECIFIED
69[a]	INITIATED		ROUTED	
70	RECEIVED		NONE	
71	RECEIVEDTRANSFER		NONE	
72	RECEIVEDCONSULT		NONE	
73	ROUTEDTO		NONE	
74	DIVERTEDTO		NONE	
75	INITIATEDCONSULT		NONE	
76	INCONFERENCE		NONE	
77	INITIATED		NONE	
78[a]	RECEIVEDREQUEST		NONE	
79	PULLER		NONE	
80	REDIRECTEDTO		NONE	
82[a]	INCONFERENCE	CONFERENCEJOINED	ABANDONED	UNSPECIFIED
83	INCONFERENCE		CUSTOMERABANDONED	ABANDONEDWHILERINGING
84[a]	INCONFERENCE	CONFERENCEJOINED	CUSTOMERABANDONED	ABANDONEDWHILERINGING

Table 9: Supported Technical Descriptor Combinations (Continued)

TECHNICAL_DESCRIPTOR_KEY	Resource Role	Role Reason	Technical Result	Technical Result Reason
85	INITIATEDCONSULT		DESTINATIONBUSY	
86	RECEIVED		DIVERTED	ANSWEREDBYAGENT
87	RECEIVED		DIVERTED	ANSWEREDBYOTHER
88	RECEIVED		DIVERTED	REDIRECTED
89	RECEIVED		DIVERTED	ABANDONEDWHILERINGING
90	RECEIVED		REDIRECTED	ROUTEONNOANSWER
91	ROUTEDTO		REDIRECTED	ROUTEONNOANSWER
92	DIVERTEDTO		REDIRECTED	ROUTEONNOANSWER
93	RECEIVEDCONSULT		REDIRECTED	ROUTEONNOANSWER
94	RECEIVEDTRANSFER		REDIRECTED	ROUTEONNOANSWER
95	INCONFERENCE		REDIRECTED	ROUTEONNOANSWER
96	RECEIVED		REDIRECTED	
97	ROUTEDTO		REDIRECTED	
98	DIVERTEDTO		REDIRECTED	
99	RECEIVEDCONSULT		REDIRECTED	
100	RECEIVEDTRANSFER		REDIRECTED	
101	INCONFERENCE		REDIRECTED	

Table 9: Supported Technical Descriptor Combinations (Continued)

TECHNICAL_DESCRIPTOR_KEY	Resource Role	Role Reason	Technical Result	Technical Result Reason
102	RECEIVED		CLEARED	ROUTEDFROMANOTHERVQ
103	RECEIVED		CLEARED	DEFAULTROUTEDBYSTRATEGY
104	RECEIVED		CLEARED	DEFAULTROUTEDBYSWITCH
105	RECEIVED		CLEARED	TARGETSCLEARED
106	RECEIVED		CONFERENCED	
107	RECEIVEDTRANSFER		CONFERENCED	
108	ROUTEDTO		CONFERENCED	
109	DIVERTEDTO		CONFERENCED	
110	INCONFERENCE		CONFERENCED	
111	RECEIVEDCONSULT		DIVERTED	ABANDONEDWHILERINGING
112	RECEIVEDCONSULT		DIVERTED	ANSWEREDBYAGENT
113	RECEIVEDCONSULT		DIVERTED	ANSWEREDBYOTHER
114	RECEIVEDCONSULT		DIVERTED	REDIRECTED
115	RECEIVEDCONSULT		DIVERTED	ROUTEONNOANSWER
116[a]	ROUTEDTO		REDIRECTED	REJECTED
117[a]	ROUTEDTO		REDIRECTED	REVOKED
118	RECEIVEDTRANSFER		ABANDONED	REJECTED

Table 9: Supported Technical Descriptor Combinations (Continued)

TECHNICAL_DESCRIPTOR_KEY	Resource Role	Role Reason	Technical Result	Technical Result Reason
119	RECEIVEDCONSULT		ABANDONED	REJECTED
120	INCONFERENCE		ABANDONED	REJECTED
121	RECEIVEDTRANSFER		ABANDONED	REVOKED
122	RECEIVEDCONSULT		ABANDONED	REVOKED
123	INCONFERENCE		ABANDONED	REVOKED
124	INITIATED		REDIRECTED	PULLEDBACK (starting with release 8.1.4) or PULLEDBACKTIMEOUT (releases earlier than 8.1.4)
125	PULLER		REDIRECTED	PULLEDBACK (starting with release 8.1.4) or PULLEDBACKTIMEOUT (releases earlier than 8.1.4)
126[a]	REDIRECTEDTO	PULLEDBACK (starting with release 8.1.4) or PULLEDBACKTIMEOUT (releases earlier than 8.1.4)	NONE	UNSPECIFIED
127[a]	REDIRECTEDTO	PULLEDBACK (starting with release 8.1.4) or PULLEDBACKTIMEOUT (releases earlier than 8.1.4)	ABANDONED	UNSPECIFIED
128[a]	REDIRECTEDTO	PULLEDBACK (starting with release 8.1.4) or PULLEDBACKTIMEOUT (releases earlier than 8.1.4)	CUSTOMERABANDONED	UNSPECIFIED

Table 9: Supported Technical Descriptor Combinations (Continued)

TECHNICAL_DESCRIPTOR_KEY	Resource Role	Role Reason	Technical Result	Technical Result Reason
129[a]	REDIRECTEDTO	PULLEDBACK (starting with release 8.1.4) or PULLEDBACKTIMEOUT (releases earlier than 8.1.4)	CUSTOMERABANDONED	ABANDONEDWHILEQUEUED
130[a]	REDIRECTEDTO	PULLEDBACK (starting with release 8.1.4) or PULLEDBACKTIMEOUT (releases earlier than 8.1.4)	PULLED	UNSPECIFIED
131	RECEIVED		CUSTOMERABANDONED	ANSWEREDBYOTHER
132	RECEIVEDTRANSFER		CUSTOMERABANDONED	ANSWEREDBYOTHER
133	ROUTEDTO		CUSTOMERABANDONED	ANSWEREDBYOTHER
134	DIVERTEDTO		CUSTOMERABANDONED	ANSWEREDBYOTHER
135	RECEIVED		CLEARED	PULLEDBACK (starting with release 8.1.4) or PULLEDBACKTIMEOUT (releases earlier than 8.1.4)
136	RECEIVED		CLEARED	STOPPED
137	RECEIVED		DIVERTED	REVOKED
138	RECEIVED		DIVERTED	REJECTED
139	RECEIVED		DIVERTED	ROUTEDTOOTHER
140[a]	INCONFERENCE	CONFERENCEJOINED	REDIRECTED	ROUTEONNOANSWER

Table 9: Supported Technical Descriptor Combinations (Continued)

TECHNICAL_DESCRIPTOR_KEY	Resource Role	Role Reason	Technical Result	Technical Result Reason
141[a]	INCONFERENCE	CONFERENCEJOINED	REDIRECTED	UNSPECIFIED
142	INITIATED		CONFERENCED	UNSPECIFIED
143	PULLER		PULLED	UNSPECIFIED
144[a]	INCONFERENCE	CONFERENCEINITIATOR	TRANSFERRED	UNSPECIFIED
145[a]	INCONFERENCE	CONFERENCEJOINED	TRANSFERRED	UNSPECIFIED
146[a]	RECEIVEDTRANSFER		REDIRECTED	REJECTED
147[a]	RECEIVEDTRANSFER		REDIRECTED	REVOKED
200	RECEIVEDCONSULT		CONFERENCED	UNSPECIFIED
201	INITIATED		CUSTOMERABANDONED	ABANDONEDFROMHOLD
202	UNKNOWN		ABANDONED	REVOKED
203	RECEIVEDTRANSFER		OUTBOUNDSTOPPED	UNSPECIFIED
204	ROUTEDTO		OUTBOUNDSTOPPED	UNSPECIFIED
205	INITIATED		OUTBOUNDSTOPPED	UNSPECIFIED
206	PULLER		OUTBOUNDSTOPPED	UNSPECIFIED
207	UNKNOWN		ABANDONED	ABANDONEDWHILEQUEUED
208	UNKNOWN		ABANDONED	REDIRECTED
209	UNKNOWN		ABANDONED	UNSPECIFIED

Table 9: Supported Technical Descriptor Combinations (Continued)

TECHNICAL_DESCRIPTOR_KEY	Resource Role	Role Reason	Technical Result	Technical Result Reason
210	UNKNOWN		CLEARED	DEFAULTROUTEDBYSTRATEGY
211	UNKNOWN		CLEARED	DEFAULTROUTEDBYSWITCH
212	UNKNOWN		CLEARED	PULLEDBACK (starting with release 8.1.4) or PULLEDBACKTIMEOUT (releases earlier than 8.1.4)
213	UNKNOWN		CLEARED	ROUTEDFROMANOTHERVQ
214	UNKNOWN		CLEARED	STOPPED
215	UNKNOWN		CLEARED	STUCKCALL
216	UNKNOWN		CLEARED	TARGETSCLEARED
217	UNKNOWN		CLEARED	UNSPECIFIED
218	UNKNOWN		COMPLETED	UNSPECIFIED
219	UNKNOWN		CONFERENCED	UNSPECIFIED
220	UNKNOWN		CUSTOMERABANDONED	ABANDONEDFROMHOLD
221	UNKNOWN		CUSTOMERABANDONED	ABANDONEDWHILEQUEUED
222	UNKNOWN		CUSTOMERABANDONED	ABANDONEDWHILERINGING
223	UNKNOWN		CUSTOMERABANDONED	ANSWEREDBYOTHER
224	UNKNOWN		CUSTOMERABANDONED	UNSPECIFIED
225	UNKNOWN		DESTINATIONBUSY	UNSPECIFIED

Table 9: Supported Technical Descriptor Combinations (Continued)

TECHNICAL_DESCRIPTOR_KEY	Resource Role	Role Reason	Technical Result	Technical Result Reason
226	UNKNOWN		DIVERTED	ABANDONEDWHILERINGING
227	UNKNOWN		DIVERTED	ANSWEREDBYAGENT
228	UNKNOWN		DIVERTED	ANSWEREDBYOTHER
229	UNKNOWN		DIVERTED	REDIRECTED
230	UNKNOWN		DIVERTED	REJECTED
231	UNKNOWN		DIVERTED	REVOKED
232	UNKNOWN		DIVERTED	ROUTEDTOOTHER
233	UNKNOWN		DIVERTED	ROUTEONNOANSWER
234	UNKNOWN		DIVERTED	UNSPECIFIED
235	UNKNOWN		OUTBOUNDSTOPPED	UNSPECIFIED
236	UNKNOWN		PULLED	UNSPECIFIED
237	UNKNOWN		REDIRECTED	PULLEDBACK (starting with release 8.1.4) or PULLEDBACKTIMEOUT (releases earlier than 8.1.4)
238[a]	UNKNOWN		REDIRECTED	REJECTED
239[a]	UNKNOWN		REDIRECTED	REVOKED
240	UNKNOWN		REDIRECTED	ROUTEONNOANSWER
241	UNKNOWN		REDIRECTED	UNSPECIFIED

Table 9: Supported Technical Descriptor Combinations (Continued)

TECHNICAL_DESCRIPTOR_KEY	Resource Role	Role Reason	Technical Result	Technical Result Reason
242[a]	UNKNOWN		ROUTED	UNSPECIFIED
243	UNKNOWN		TRANSFERRED	UNSPECIFIED
244	INCONFERENCE		CUSTOMERABANDONED	ABANDONEDWHILEQUEUED
245	RECEIVEDCONSULT		CLEARED	UNSPECIFIED
246	RECEIVEDCONSULT		CLEARED	STUCKCALL
247	RECEIVEDCONSULT		CLEARED	ROUTEDFROMANOTHERVQ
248	RECEIVEDCONSULT		CLEARED	DEFAULTROUTEDBYSTRATEGY
249	RECEIVEDCONSULT		CLEARED	DEFAULTROUTEDBYSWITCH
250	RECEIVEDCONSULT		CLEARED	TARGETSCLEARED
251	RECEIVEDCONSULT		CLEARED	PULLEDBACK (starting with release 8.1.4) or PULLEDBACKTIMEOUT (releases earlier than 8.1.4)
252	RECEIVEDCONSULT		CLEARED	STOPPED
253	UNKNOWN		ABNORMALSTOP	
254	RECEIVED		ABNORMALSTOP	
255	RECEIVEDTRANSFER		ABNORMALSTOP	
256	ROUTEDTO		ABNORMALSTOP	
257	INCONFERENCE		ABNORMALSTOP	

Table 9: Supported Technical Descriptor Combinations (Continued)

TECHNICAL_DESCRIPTOR_KEY	Resource Role	Role Reason	Technical Result	Technical Result Reason
258	INITIATED		ABNORMALSTOP	
259	PULLER		ABNORMALSTOP	
260	REDIRECTEDTO		ABNORMALSTOP	
261	UNKNOWN		ABNORMALSTOP	ABNORMALSTOPWHILEQUEUED
262	UNKNOWN		ABNORMALSTOP	ABNORMALSTOPWHILERINGING
263	RECEIVED		ABNORMALSTOP	ABNORMALSTOPWHILEQUEUED
264	RECEIVED		DIVERTED	ABNORMALSTOPWHILERINGING
265	RECEIVEDTRANSFER		ABNORMALSTOP	ABNORMALSTOPWHILEQUEUED
266	RECEIVEDTRANSFER		ABNORMALSTOP	ABNORMALSTOPWHILERINGING
267	ROUTEDTO		ABNORMALSTOP	ABNORMALSTOPWHILEQUEUED
268	ROUTEDTO		ABNORMALSTOP	ABNORMALSTOPWHILERINGING
269	INCONFERENCE		ABNORMALSTOP	ABNORMALSTOPWHILERINGING
270	PULLER		ABNORMALSTOP	ABNORMALSTOPWHILEQUEUED
271	REDIRECTEDTO		ABNORMALSTOP	ABNORMALSTOPWHILEQUEUED
272	UNKNOWN		COMPLETED	ARCHIVED
273	RECEIVED		COMPLETED	ARCHIVED
274	RECEIVEDTRANSFER		COMPLETED	ARCHIVED

Table 9: Supported Technical Descriptor Combinations (Continued)

TECHNICAL_DESCRIPTOR_KEY	Resource Role	Role Reason	Technical Result	Technical Result Reason
275	ROUTEDTO		COMPLETED	ARCHIVED
276	INITIATED		COMPLETED	ARCHIVED
277	PULLER		COMPLETED	ARCHIVED
278	REDIRECTEDTO		COMPLETED	ARCHIVED
279	UNKNOWN		COMPLETED	CANCELED
280	RECEIVED		COMPLETED	CANCELED
281	RECEIVEDTRANSFER		COMPLETED	CANCELED
282	ROUTEDTO		COMPLETED	CANCELED
283	INITIATED		COMPLETED	CANCELED
284	PULLER		COMPLETED	CANCELED
285	REDIRECTEDTO		COMPLETED	CANCELED
286	INCONFERENCE		REDIRECTED	PULLEDBACK
287	ROUTEDTO		REDIRECTED	PULLEDBACK

a. Genesys Info Mart 8.1 does not use this technical descriptor dimension.

Genesys

Supplements

Related Documentation Resources

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

Framework

- The *Framework 8.x Management Layer User's Guide* provides information about the concepts, terminology, and procedures that apply to this layer of the Genesys Framework.
- The *Framework 8.x Configuration Options Reference Manual* provides information about configuration options for Framework components.
- The *Framework 8.x Configuration Manager Help* provides information about using Configuration Manager in either an enterprise or a multi-tenant environment.
- The *Framework 8.x Deployment Guide* provides information about configuring, installing, starting, and stopping Framework components.
- The *Framework 8.x Combined Log Events Help* describes log events that Genesys server applications generate and that Solution Control Interface displays. The *Framework 8.x Combined Log Events Help* includes descriptions of Genesys Info Mart log events.

Interaction Concentrator

- The *Interaction Concentrator 8.x Deployment Guide* provides information about architecture, configuration requirements, and installation steps for Interaction Concentrator, and it describes how to make data from the Genesys Outbound Contact solution available in Interaction Database (IDB).

- The *Interaction Concentrator 8.x User's Guide* provides basic information about IDB architecture and detailed information about Interaction Concentrator features and functionality, including attached data processing, available stored procedures, and integration with other Genesys products.
- The *Interaction Concentrator 8.x Physical Data Model* for your relational database management system (RDBMS) provides information about the IDB schemas.

Genesys Info Mart

- The *Genesys Info Mart 8.1 Deployment Guide* provides information about architecture, configuration requirements, and installation steps for Genesys Info Mart, Genesys Info Mart Manager, and the Genesys Info Mart Administration Console. This guide also provides in-depth information about data processing, maintenance, and purging.
- The *Genesys Info Mart 8.1 Operations Guide* provides information about the Genesys Info Mart jobs. The jobs execute extract, transform, and load (ETL) processes, maintain the Info Mart database, and migrate the database schema as required. The guide also explains how to use the Genesys Info Mart Administration Console to monitor and administer the jobs.
- The *Genesys Info Mart8.1 Reference Manual* for your RDBMS provides information about the Info Mart database schema.
- The *Genesys Info Mart 8.1 Business Continuity Deployment Guide*, which is available as wiki pages, provides information and procedures that are relevant to Genesys Info Mart deployment in an environment that requires support for Business Continuity.
- The *Genesys Info Mart 8.1 Database Size Estimator* helps you estimate the size of your Info Mart database when you are planning your deployment. The estimator is a Microsoft Office Excel 2007 spreadsheet that is available from the Genesys Documentation website.
- The *Genesys Info Mart Database Compatibility Reference* includes compatibility information for database tables and fields that existed in the Genesys Info Mart database schema in release 7.6. This document, which is available as wiki pages, provides guidelines for mapping Info Mart 7.6 database SQL queries for use with an Info Mart 8.x database.
- Release Notes and Product Advisories for this product, which are available on the Genesys Customer Care website at http://genesys.com/customer-care

Reporting and Analytics Aggregates

- The *Reporting and Analytics Aggregates 8.1 Deployment Guide* describes how to deploy the Reporting and Analytics Aggregates (RAA) package provided with Genesys Info Mart.
- The *Reporting and Analytics Aggregates 8.1 Reference Manual* describes the aggregate tables that are available to Genesys Info Mart customers with deployment of RAA.
- The *Reporting and Analytics Aggregates 8.1 User's Guide* describes the aggregation process, provides the aggregation hierarchies, and explains how to enable aggregation of user data.

Genesys Interactive Insights

- The *Genesys Interactive Insights 8.1 Deployment Guide* describes how to install Genesys Interactive Insights (GI2) and set up the environment required in order to run the GI2 reports.
- The *Genesys Interactive Insights 8.1 Universe Guide* describes, in detail, the reports and measures that are provided in the GI2 release.
- The *Genesys Interactive Insights 8.1 User's Guide* summarizes how to operate GI2 reports, provides basic instructions for customizing your own reports, and provides information about the report upgrade utility.

Genesys

- The *Genesys Technical Publications Glossary,* which is 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.
- The *Genesys Migration Guide*, which ships on the Genesys Documentation Library DVD, provides documented migration strategies for Genesys product releases. The Genesys Info Mart 8.x part of the guide includes instructions on how to migrate Genesys Info Mart from release 8.0.x to release 8.1. Contact Genesys Customer Care for more information.

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

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

Consult the following additional resources as necessary:

- The *Genesys Hardware Sizing Guide* provides information about Genesys hardware sizing guidelines for the Genesys 8.x releases.

- The *Genesys Interoperability Guide* provides information on the compatibility of Genesys products with various Configuration Layer Environments; Interoperability of Reporting Templates and Solutions; and *Gplus* Adapters Interoperability.
- The *Genesys Licensing Guide* introduces you to the concepts, terminology, and procedures that are relevant to the Genesys licensing system.
- The *Genesys Database Sizing Estimator 8.x Worksheets* 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 Customer Care 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 10 describes and illustrates the type conventions that are used in this document.

Table 10: 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 198).	Please consult the *Genesys Migration Guide* for more information. Do *not* use this value for this option. A *customary and usual* practice is one that is widely accepted and used within a particular industry or profession. The formula, $x + 1 = 7$ where x stands for . . .

Table 10: Type Styles (Continued)

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

Index

Symbols

A

B

C

D

E

F

G

H

I

J

L

M

N

O

P

Q

R

S

T

U

V

W