INTELLIGENT WORKLOAD DISTRIBUTION DATA MART

Reference Guide

May 2009
iWD DM 7.6.1

Copyright

Disclaimer

Feedback

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

Trademarks and Registered Trademarks

Revision History

Version	Issued	Description
1.4.1	18.06.2008	First Draft Release for GTL release 1.4
1.4.2	30.07.2008	Second Draft Release for GTL 1.4
1.4.3	23.09.2008	Final Release for GTL 1.4.14
1.6.1	08.04.2009	Final Release for GTL 1.6.6
7.6.1.1	16.04.2009	Rebranding to iWD 7.6.1
7.6.1.2	20.05.2009	Final release for iWD 7.6.1

TABLE OF CONTENTS

Preface

Welcome to the *Reference Guide* for *intelligent Workload Distribution Data Mart.* This document introduces you to the schemas that compose the intelligent Workload Distribution Data Mart, or iWD Data Mart, to guide you in the design and creation of reports using the data within the iWD Data Mart.

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

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

In brief, you will find the following information about iWD Data Mart in this reference guide:

- iWD and iWD Reporting Overview
- iWD Data Mart Facts, Dimensions, Aggregates, and Views
- iWD ETL Jobs

Intended Audience

This guide is intended for reporting and business analysts who want to leverage the data that is contained in the iWD Data Mart to produce reports and dashboards for business users. It is also intended for IT Administrators who would like to gain an understanding of the components that enable iWD Data Mart.

This reference guide assumes that the reader has an understanding of relational database concepts, Structured Query Language (SQL) for analyzing and mining the data, iWD, and data warehouse concepts - including working with star schemas, aggregates, and ETL concepts.

Recommended Reading

Genesys iWD Data Mart uses Kettle, a third-party ETL open-source product from Pentaho, the leader in open-source business intelligence. For more information on Kettle, please visit www.pentaho.com.

In addition, the reader is encouraged to read iWD product documentation to gain an understanding of the concepts in iWD, particularly an emphasis on the various task states that are used heavily in reporting and in the creation of many iWD Data Mart measures.

As comprehensive task and interaction reports can be made available by leveraging data from iWD Data Mart and Genesys Info Mart, the reader is encouraged to review the product and reference guide documentation, specifically the *Genesys Info Mart User's Guide* and the *Reference Guides* that are provided by your Genesys account team or partner.

Chapter Summaries

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

- Chapter 1: iWD Reporting
- Chapter 2: iWD Data Mart
- Chapter 3: ETL Overview

Document Conventions

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

Type Styles

Italic

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

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

Monospace Font

A monospace font, which resembles `teletype` or `typewriter` text, is used for all programming identifiers and graphical user interface (GUI) elements. This convention includes the names of directories, files, folders, configuration objects, paths, scripts, dialog boxes, options, fields, text and list boxes, operational modes, all buttons (including radio buttons), check boxes, commands, tabs, CTI events, and error messages; the values of options; logical arguments and command syntax; and code samples. For example:

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

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

Screen Captures in This Document

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

Square Brackets

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

Angle Brackets

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

Related Resources

Consult these additional resources, as necessary:

- iWD Online Help
- *iWD User Guide*

Chapter 1: iWD Reporting

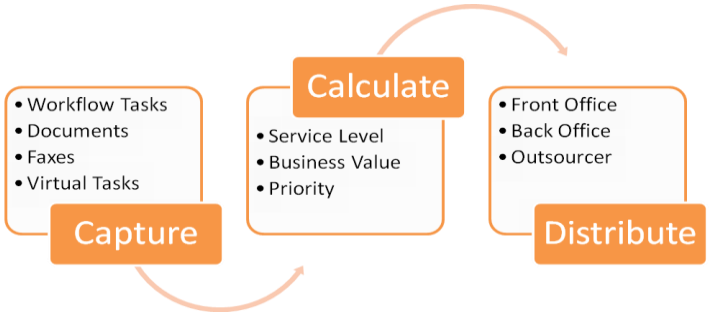
This chapter explains the reporting functionality that is available within *intelligent Workload Distribution*. The information in this chapter is organized in the following topics:

What Is intelligent Workload Distribution?

Designed as a business application for business users, Genesys intelligent Workload Distribution (iWD), works in concert with telephony and unified communication applications - enabling a centralized service-delivery platform, and proactively managing interactions and tasks across all channels and media.

Only with a global task list, sorted based on business value, can the enterprise ensure that the right resources - regardless of location - are proactively receiving the most critical or highest value tasks - regardless of media type - at the right time. The abilities to react quickly with new business intelligence and to work effectively are key success factors with any enterprise in today's competitive marketplace.

iWD supports three main areas: capturing tasks, calculating task values, and distributing tasks.



Task Capture

To address the challenge of tasks stored across multiple enterprise systems, iWD accepts work - in electronic format - from a broad range of applications, such as customer relationship management, workflow, host systems, and enterprise service bus systems.

Calculate

By using business rules that are configured by business users, iWD calculates service-level values like task due date, business value, and priority. By using these values,

iWD orders tasks from most to least important, and monitors and proactively manages tasks to ensure compliance with service-level objectives.

Distribute

iWD distributes tasks to front - or back-office resources, or to external partners like business process outsourcers, working in concert with existing call center applications, such as the Genesys Dynamic Contact Center and Unified Communication applications.

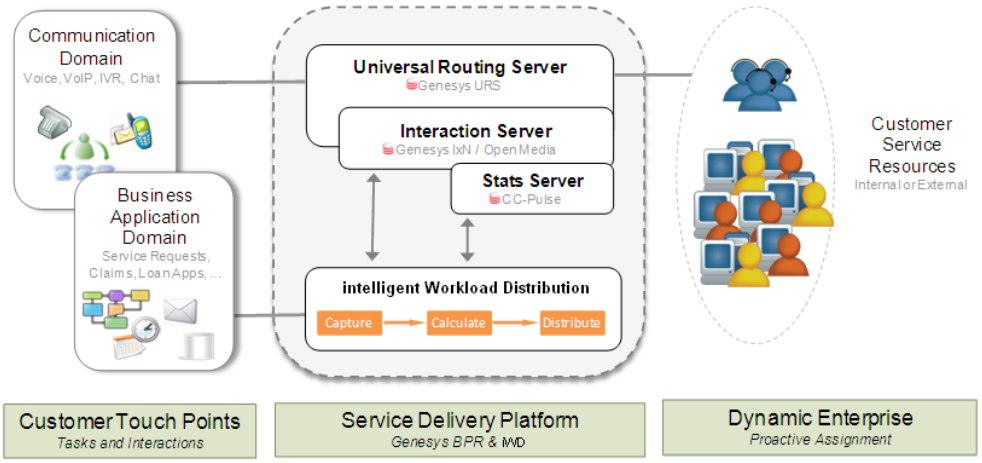
Enabling Genesys BPR Solutions

Genesys iWD enables Genesys BPR solutions, providing out-of-the-box functionality designed for the business user and with the business context organizations, while eliminating the need for expensive and risky custom development.

As a Genesys InterWorks Application, iWD integrates using Genesys Open Media to submit tasks by using a iWD Distribution Point. The Genesys Distribution Point and functionality within Genesys Universal Router enable blending of tasks that are submitted with voice calls and routing to agents in a common manner. The configuration of a distribution point does not require any complex coding and is a configuration of key parameters, such as Genesys Open Media Queue, that will receive the distributed task from iWD.

The distribution adapter for Genesys Open Media is fully bidirectional, ensuring that any task update or any attribute that is attached to the task interaction in Genesys, such as result code, can be returned and recorded for use in historical reporting and trend analysis.

In addition to the Open Media integration, iWD provides a real-time statistics interface to Genesys Statistics Server, providing call-center managers a single view of interactions and tasks in CCPulse+, to ensure a single application providing relevant and timely statistics on agent and operational performance in real time. More information on this interface and functionality can be found in the *iWD User Guide*.



For historical reporting, iWD Data Mart and Genesys Info Mart work in concert to provide the enterprise with comprehensive management information for intraday and historical reporting and trend analysis. Further details on iWD Data Mart are provided

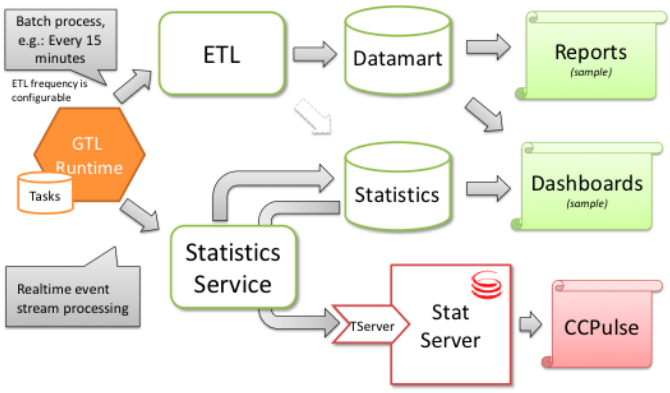
in this document, and the reader is encouraged to consult the *Genesys Info Mart Reference Guide* for further details on this Genesys product offering.
Clients who have adopted iWD to enable their Genesys BPR Solution have realized shorter time to market, lower project risk, and increased level of functionality when compared to custom development.

iWD Reporting

As customers have increased choice in the marketplace and have higher expectations of service quality, the ability to measure the efficiency and effectiveness of customer service delivery is a key component of success. iWD provides the harmony to an often-cumbersome reporting process through:

- Cradle-to-grave reporting from the time that the fax, e-mail or document is received to its completion.
- Consolidated reporting across the various systems that are involved in customer service delivery – fax servers, workflow, customer relationship management, and Genesys Customer Interaction Management.
- Reporting that is based on a business context with business process, customer segment, and product independent of channel, instead of being limited to interactions, and queues, channels, and workflows.

Key to achieving business results is having access to actionable business intelligence. Genesys iWD offers a comprehensive reporting functionality into business operations and performance against metrics and key performance indicators - in both real time through Real-Time Statistics and Historical, based on scheduled ETL process from Runtime to iWD Data Mart. The following diagram provides a functional overview of iWD Reporting components, with details provided in the sections that follow:



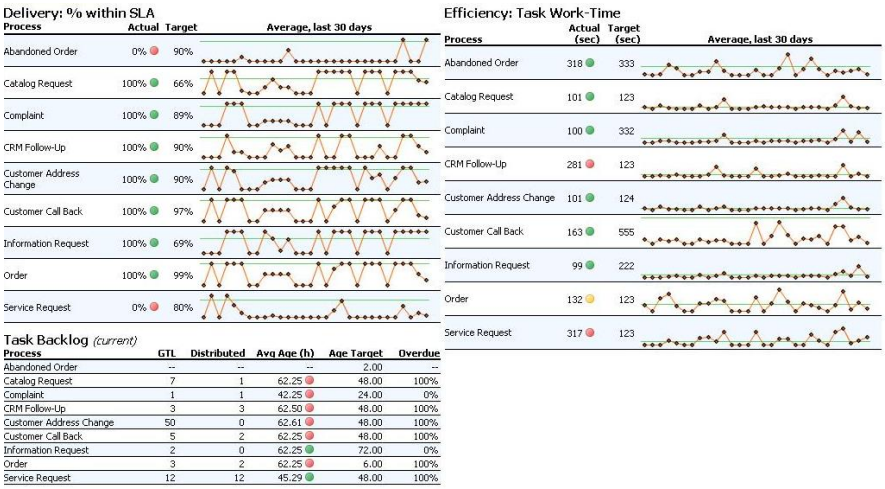
iWD Reporting Overview

Please note that each iWD tenant solution requires its own Data Mart database.

Historical& Analytics

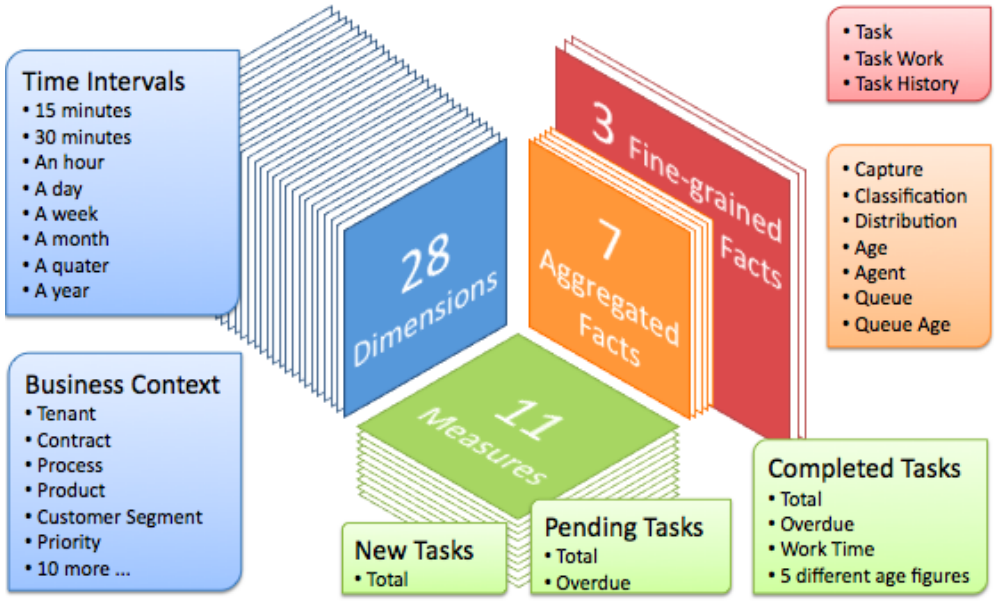
iWD Data Mart enables advanced reports and dashboards, providing users the ability to customize reports via report parameters and filters. Managers gain insight through dashboards, which offer a summary on performance against key performance

indicators and an ability to view performance trends over time. The sample iWD Dashboard widgets that are provided can be incorporated into existing dashboards - providing a unified view of organization performance from one portal - or can be utilized as a guide for creating additional dashboard widgets to suit the specific needs of your operation.



Sample iWD Dashboards

The information that is supplied in the dashboards above is retrieved from a comprehensive set of star schemas, which are designed specifically with a task-oriented business context. iWD Data Mart provides business analysts with summaries and aggregates that are required in support of business strategies through a combination of measures, fine-grained and aggregate fact tables, and dimensions, as shown below:



iWD DataMart – Dimensions, Meausres and Facts

Not unlike other data warehouse, iWD Data Mart consists of the following:

- Dimensions – Describe task attributes that are common across the Fact tables in iWD Data Mart, such as iWD Business Process, Priority, Business Value, Date, and Time. Fact tables link to dimensions through keys.
- Aggregated Fact tables – Describe tasks in a iWD-oriented context across the various stages, or the iWD life cycle of the task from capture, classification, and distribution to agent and, of course, the age of tasks. These Fact tables are at an aggregate level, with each containing a number of measures.
- Measures – Represent numerical values such as total, average, minimum, or maximum, that are stored in an Aggregated Fact table across intraday and historical intervals. For example, the total number of completed tasks by 15-minute interval by iWD Process and Business Value would be measure in the intraday Capture Aggregate Fact table.
- Fine-Grained Fact tables – Store all task attributes that are associated with the task (Task Fact table), any work-related events (Task Work Fact table) when the task was assigned to one or more agents, and task full audit history of the task (Task History Fact table).

When they are connected to existing enterprise DataMarts, including Genesys Info Mart, analysts gain access to comprehensive views of the entire customer experience. Analytical reporting leverages existing business-intelligence tools such as those that are provided by Pentaho, which is an open-source product suite for business intelligence, or through a host of commercial products from Cognos or Business Objects Crystal Reports.

iWD Extended Statistics

Statistics are generated by Kettle plug-ins which are executed by Scheduled ETL service with Aggregate Statistics job.

Aggregate Statistics job reads `stats.properties` file and executes all listed plug-ins providing them 3 parameters:

- Solution ID
- Statistics Service ID
- Time interval key

Using these parameters plug-in can query facts, aggregated facts and dimensions to create required statistics.

There are 5 sample plug-ins provided in standard package:

Filename	Description
contract_activeheld.ktr	Calculates amount of tasks in Active and Held states for each contract
contract_newcompleted.ktr	Calculates amount of new and completed tasks in last 15, 30 and 60 minutes for each contract
process_activeheld.ktr	Calculates amount of tasks in Active and Held states for each business process
process_pendingoverdue.ktr	Calculates amount of pending and overdue tasks for each business process
solution_newcompleted.ktr	Calculates amount of new and completed tasks in last 15, 30 and 60 minutes for current solution

There are embedded notes in `process_pendingoverdue.ktr` for each step plug-in makes to compute statistics.

Statistics are written to DataMart database `gtl_stats` table.

Structure of `gtl_stats` table:

Attribute	MySQL Type	Oracle Type	Description
ID	INTEGER	INTEGER	Primary key
TENANTID	VARCHAR(16)	VARCHAR2(16)	iWD Manager ID of the tenant
SOLUTIONID	VARCHAR(16)	VARCHAR2(16)	iWD Manager ID of the solution
STATSERVICEID	VARCHAR(16)	VARCHAR2(16)	iWD Manager ID of Extended statistics service used to submit statistics to Genesys
DIMENSIONTYPE	VARCHAR(32)	VARCHAR2(32)	Dimension type (SLT - solution, CRT - contract, PRC - process)
DIMENSIONID	VARCHAR(64)	VARCHAR2(64)	Dimension ID. Used to generate Genesys statistics filters.
DIMENSIONNAME	VARCHAR(255)	VARCHAR2(255)	Not used by Extended statistics
MEASUREID	VARCHAR(64)	VARCHAR2(64)	Measure ID. Used to generate Genesys statistical types.
MEASURENAME	VARCHAR(255)	VARCHAR2(255)	Not used by Extended statistics
MEASUREVALUE	INTEGER	INTEGER	Measure value
ACTIVATIONDATETIME	DATETIME	TIMESTAMP	Not used by Extended statistics

Extended statistics service reads `gtl_stats` table and submits statistics to Genesys using Genesys Statistics Adapter service.

Extended statistics service is using in-memory cache to store submitted statistics. Only statistics for new and updated dimensions are submitted to Genesys Statistics Adapter service.

Aggregate Statistics job does not create audit event in `ETL_AUDIT` table.

iWD Task Attributes

At the heart of iWD Reporting is a set of task attributes that describe the task and its association with the iWD business context, such as Business Process. Task attributes span across three categories:

- Core Attributes
- Extended Attributes
- Custom Attributes

An understanding of these attributes is a key to understanding the measures and aggregates in iWD Data Mart. Incorrect use or interpretation of these values can negatively affect reporting outcomes. Please consult the iWD User Guides for further information on these attributes.

Core Attributes

Core attributes are at the heart of intelligent Workload Distribution in that they describe the fundamentals of a task and are used in assembling tasks in the global list, based on iWD Business Value and iWD Priority. These values are set automatically by iWD, or through iWD business rules, or they can be provided by the source system through iWD Web Service, MQ Series or a Capture adapter interface. Examples include iWD Process ID, Business Value, and Priority.

Extended Attributes

These attributes are intended to provide additional task context - allowing customers to tailor the SLA rules for managing tasks on the global task list - and can aid in customizing real-time and historical reporting. For example, use of several capture dates allows an organization to measure organization performance against the date/time that an order or loan was received by the originating system or was submitted via a web form by the customer.

Custom Attributes

In addition to the Core and Extended Attributes, iWD provides customers with the ability to submit additional task details through iWD Custom Attributes. These are key/value pairs and are provided by the source system. For example, a web form may contain several fields that not already mapped to one of the Core or Extended Attributes.

When they are submitted via a iWD Capture adapter such as iWD Web Service, XML file capture, or MQ Series, the values are stored in the iWD runtime database. They can also be mapped to Custom Dimensions in iWD Data Mart by defining mapping properties on the ETL configuration in iWD Manager – again, further extending the level of tailored reporting that is enabled by iWD Data Mart. For more information on how to utilize custom dimensions, please consult the iWD User Guide.

Chapter 2: iWD Data Mart

Genesys iWD Data Mart comprises a set of star schemas that consist of Fact and Dimension tables. In a star schema, facts are the central table that contains the individual facts about a task. There are two types of fields in a fact table:

- Fields that store the foreign keys that connect each particular fact to the appropriate value in each dimension.
- Fields that store the individual facts (or measures), such as number, amount, or price.

There are three types of facts in iWD Data Mart, with data from intraday moved to historical on a 24-hour basis:

- Intraday – Facts and aggregations for current day, used for near-real-time dashboards and reports
- Historical – Historical facts and aggregations, used for historical and analytical reporting
- Blended – Database views that combine intraday and historical facts

Surrounding a Fact table are Dimensions. Dimensions contain attributes about a task that are common across multiple tasks, such as date, time, iWD business process, or tenant. The combination of a fact and a set of dimensions create a star schema.

This chapter provides an overview of the facts and dimensions within iWD Data Mart:

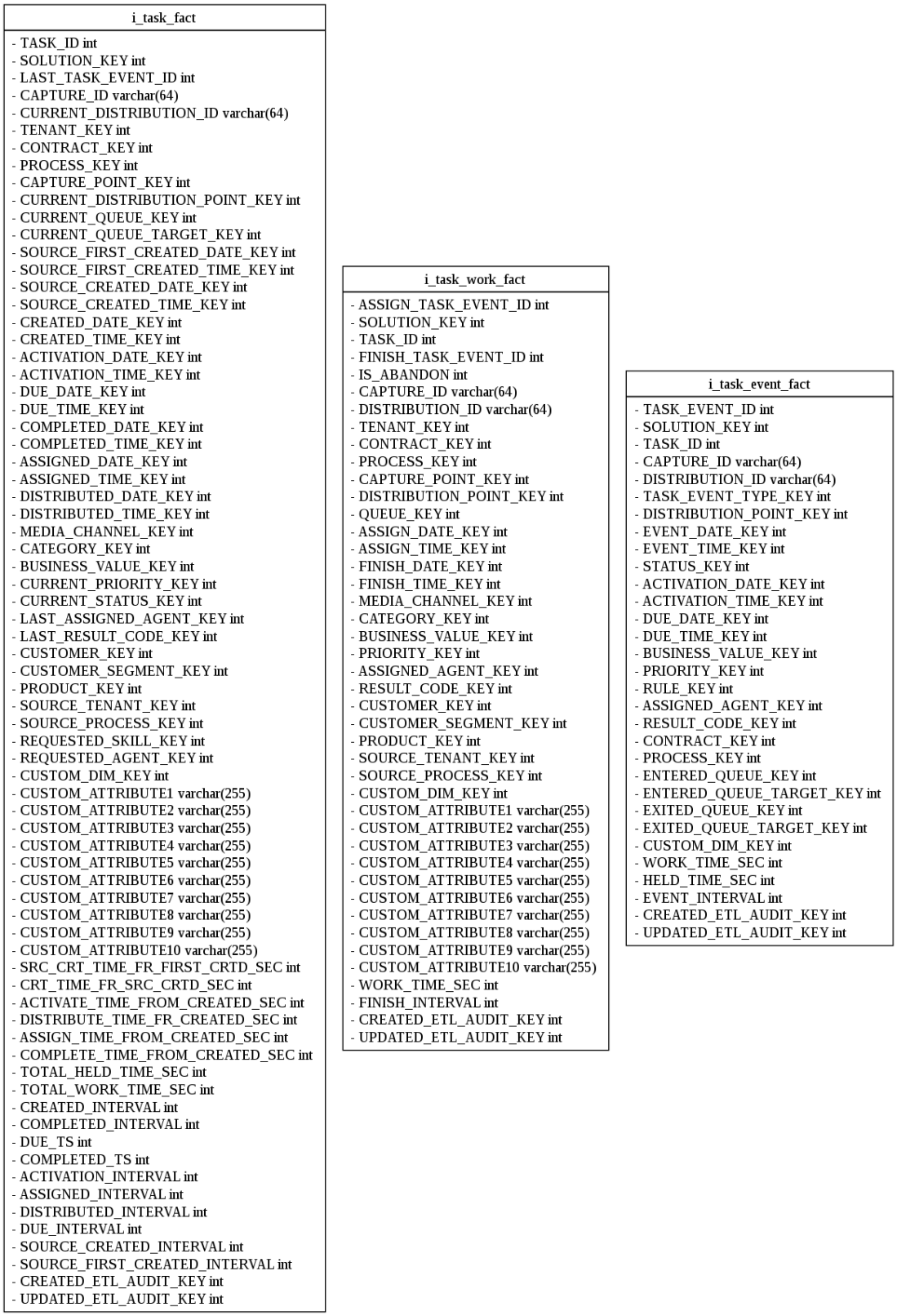
iWD Data Mart Core Facts

Task Fact (task_fact)

Task Fact describes a task from customer's perspective. Each iWD runtime task record will result in a single Task Fact. In the iWD DM database, Task Fact can be accessed via the following:

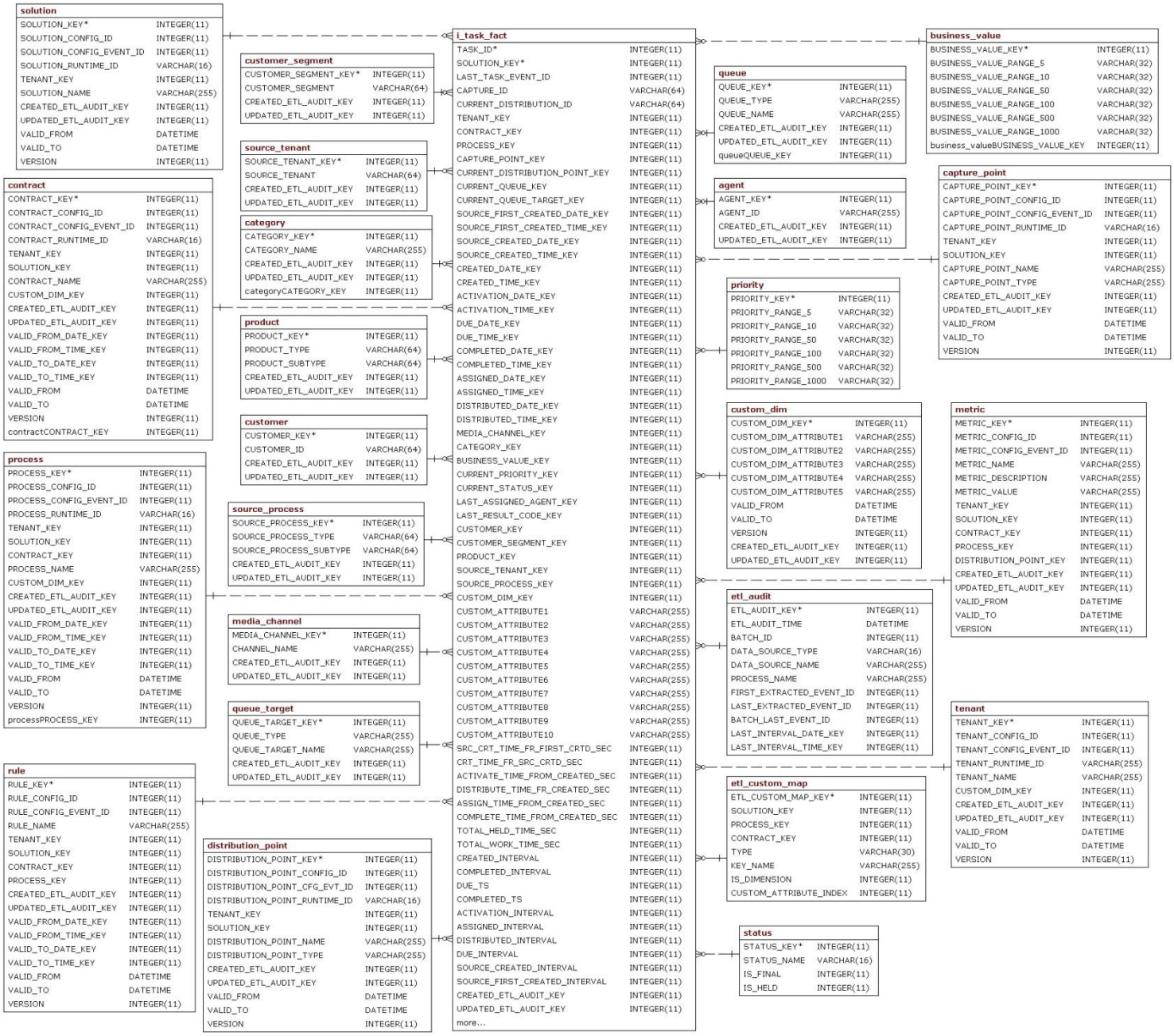
- i_task_fact - Intraday data table
- h_task_fact - Historical data table
- task_fact - Blended (historical and intraday) data view

The following is a view of the Task Facts that are listed above, with details on each provided in the tables below:



iWD DataMart Fact Tables

The following is a view of the i_task_fact star schema, with the standards dimensions surrounding the Task Fact:



The Task Fact table contains the following fields:

Attribute	MySQL Type	Oracle Type	PK	Description
TASK_ID	INTEGER	INTEGER	X	Task ID in iWD, generated automatically when new task is created, unique within a single iWD database.
SOLUTION_KEY	INTEGER	INTEGER	X	Key to the solution dimension describing the task's Solution Instance as configured in iWD Manager (for example, Production vs. Test). It is assigned as soon as task is created in IWD runtime. A tenant can have one or many solution instances.

Attribute	MySQL Type	Oracle Type	PK	Description
LAST_TASK_EVENT_ID	INTEGER	INTEGER		Unique identifier for the last event associated with the task.
CAPTURE_ID	VARCHAR(64)	VARCHAR2(64)		The ID of the task capture, as stored or referenced in the originating system - for example, work-item ID.
CURRENT_DISTRIBUTION_ID	VARCHAR(64)	VARCHAR2(64)		Distribution ID for the task as assigned by the routing system, like Genesys CIM. In the case of CIM, this would be the Genesys Interaction ID for the task, returned to iWD via the iWD Distribution Point/Open Media interface.
TENANT_KEY	INTEGER	INTEGER		Key to the tenant dimension describing the task's Tenant, as configured in iWD Manager. This is assigned as soon as task is created in iWD runtime.
CONTRACT_KEY	INTEGER	INTEGER		Key to contract dimension. Describes the task's parent iWD contract.
PROCESS_KEY	INTEGER	INTEGER		Key to process dimension. Describes the task's parent iWD business process.
CAPTURE_POINT_KEY	INTEGER	INTEGER		Key to capture point dimension. Describes the capture point that captured the task.
CURRENT_DISTRIBUTION_POINT_KEY	INTEGER	INTEGER		Key to distribution point dimension. Describes the distribution point that distributed the task.
CURRENT_QUEUE_KEY	INTEGER	INTEGER		Key to the queue dimension describing the queue the task is in and the Queue Type. For Genesys, distribution the following types are possible: • InteractionQueue. • AgentWorkbin. • AgentGroupWorkbin. • PlaceWorkbin. • PlaceGroupWorkbin. For more information on Genesys queue types listed above, please consult your Genesys user manuals and reference guides.
CURRENT_QUEUE_TARGET_KEY	INTEGER	INTEGER		Key to the target queue dimension. Queue Target is the ID of agent, agent group, place or place group to which the task is assigned. For Genesys this is only populated when queueType is AgentWorkbin, AgentGroupWorkbin, PlaceWorkbin or PlaceGroupWorkbin.
SOURCE_FIRST_CREATED_DATE_KEY	INTEGER	INTEGER		Key to date/time. Reserved for the DTM from the first system that captured the task. Two source DTM are provided with this one being reserved for cases where two systems were involved in the task origination (for example, fax server and workflow).
SOURCE_FIRST_CREATED_TIME_KEY	INTEGER	INTEGER		Time key for the first source system.
SOURCE_CREATED_DATE_KEY	INTEGER	INTEGER		Date key for the second source system. This is the system that submitted the task to iWD.
SOURCE_CREATED_TIME_KEY	INTEGER	INTEGER		Time key for the second source system submitting the task to iWD.
CREATED_DATE_KEY	INTEGER	INTEGER		Key to the iWD task created date. Additional created date/time stamps are provided for in extended attributes to report not only on iWD capture date/time, but also on the originating system (e.g.: workflow capture date/time). See Source Create and Source First Created.
CREATED_TIME_KEY	INTEGER	INTEGER		Key to the iWD task created time.
ACTIVATION_DATE_KEY	INTEGER	INTEGER		Key to the event_date dimension describing the iWD task activation date. Activation Date is the date when task becomes active, before that it will stay queued and will not be reprioritized and distributed.

Attribute	MySQL Type	Oracle Type	PK	Description
ACTIVATION_TIME_KEY	INTEGER	INTEGER		Key to the event_time dimension describing the Activation Time for the task. Activation Time is the time when task becomes active, before that it will stay queued and will not be reprioritized and distributed.
DUE_DATE_KEY	INTEGER	INTEGER		Key to the event_date dimension describing the task due date as set by iWD rules or by the source system.
DUE_TIME_KEY	INTEGER	INTEGER		Key to the event_time dimension describing task due time.
COMPLETED_DATE_KEY	INTEGER	INTEGER		Key to the event_date dimension describing the iWD completion date.
COMPLETED_TIME_KEY	INTEGER	INTEGER		Key to the event_time dimension describing task completion time.
ASSIGNED_DATE_KEY	INTEGER	INTEGER		Key to the event_date dimension describing the date when task was assigned to an agent.
ASSIGNED _TIME_KEY	INTEGER	INTEGER		Key to the event_time dimension describing the time when task was assigned to an agent.
DISTRIBUTED_DATE_KEY	INTEGER	INTEGER		Key to the event_date dimension describing the date when task was distributed.
DISTRIBUTED _TIME_KEY	INTEGER	INTEGER		Key to the event_time dimension describing the time when task was distributed.
MEDIA_CHANNEL_KEY	INTEGER	INTEGER		Key to the media_channel dimension describing channel associated with the task (for example, fax). This can be set in iWD Rules, or set by the system submitting the task.
CATEGORY_KEY	INTEGER	INTEGER		Key to category dimension. Category can be used to further describe the task, like a follow-up.
BUSINESS_VALUE_KEY	INTEGER	INTEGER		Key to the business_value dimension. Business value is assigned using iWD rules during classification phase of the task.
CURRENT_PRIORITY_KEY	INTEGER	INTEGER		Key to the priority dimension. Like business value, it is assigned during classification. Priority can change overtime. For example, as the task gets closer to its due-date, the prioritization rules may be configured to proactively reprioritize the task. This value is current priority, with history available in the task_event_fact table.
CURRENT_STATUS_KEY	INTEGER	INTEGER		Key to the current task status. The following are the iWD status and a brief description: • New: Newly created task awaiting processing. • NewHeld: Task created but will not be processed until it is resumed. • Captured: Task has been processed by iWD, but not yet prioritized. • Queued: Task is processed and prioritized at least once. • Held: Task in a held state (user action, or system), and will not be reprioritized or distributed until it is resumed. • Distributed: Task is submitted to distribution system, like Genesys Open Media queue. • Assigned: Task is assigned to an Agent. • Completed: Task is completed. • Canceled: Task is canceled. • ErrorHeld: Error has occurred during task processing, prioritization or distribution. Error details are stored in "error" custom extended task attribute. When resumed, iWD will attempt to process task again. • Rejected: Task has been rejected during processing. This can occur when the task was assigned to an expired contract or process.

Attribute	MySQL Type	Oracle Type	PK	Description
LAST_ASSIGNED_AGENT_KEY	INTEGER	INTEGER		Key to the agent dimension for the last assigned to which the task was assigned.
LAST_RESULT_CODE_KEY	INTEGER	INTEGER		Key to the result code dimension. Result Code is often the wrap code from softphone, routing strategy or from source system. It may be utilized in reporting to gain further understanding of the outcome of the task.
CUSTOMER_KEY	INTEGER	INTEGER		Key to the customer dimension. It is oftentimes used as the customer ID from source system. This ID can be utilized to retrieve further customer details from CRM data warehouse or other customer data repositories.
CUSTOMER_SEGMENT_KEY	INTEGER	INTEGER		Key to the customer segment dimension describing the customer to whom the task is associated and is received from source system as an extended iWD task attribute - for example, gold, silver or bronze.
PRODUCT_KEY	INTEGER	INTEGER		Key to the Product Dimension describing the product to which the task is related. For example, it can be a product name, or product type like a loan or Internet DSL. As Product can be further defined using product subtype such as residential loan or home DSL.
SOURCE_TENANT_KEY	INTEGER	INTEGER		Key to the source tenant dimension with the ID of the tenant submitting the task. It can be important in a multi-tenant, or service bureau environment.
SOURCE_PROCESS_KEY	INTEGER	INTEGER		Key to the source_process dimension. Source process includes type and subtype that described the source process associated with the task (for example, Order and DSL Order)
CUSTOM_DIM_KEY	INTEGER	INTEGER		Key to custom_dim dimension containing five additional attributes (beyond those listed below) that can be utilized to describe a task.
CUSTOM_ATTRIBUTE1	VARCHAR(255)	VARCHAR2(255)		Custom attributes describe a task. A total of 10 customer attributes can be mapped to the task, with 5 additional in the custom_dim dimension. If more than 10 task attributes exist, the first 10 are mapped, with the remaining left unmapped.
CUSTOM_ATTRIBUTE2	VARCHAR(255)	VARCHAR2(255)		
CUSTOM_ATTRIBUTE3	VARCHAR(255)	VARCHAR2(255)		
CUSTOM_ATTRIBUTE4	VARCHAR(255)	VARCHAR2(255)		
CUSTOM_ATTRIBUTE5	VARCHAR(255)	VARCHAR2(255)		
CUSTOM_ATTRIBUTE6	VARCHAR(255)	VARCHAR2(255)		
CUSTOM_ATTRIBUTE7	VARCHAR(255)	VARCHAR2(255)		
CUSTOM_ATTRIBUTE8	VARCHAR(255)	VARCHAR2(255)		
CUSTOM_ATTRIBUTE9	VARCHAR(255)	VARCHAR2(255)		
CUSTOM_ATTRIBUTE10	VARCHAR(255)	VARCHAR2(255)		
SRC_CRT_TIME_FR_FIRST_CRTD_SEC	INTEGER	INTEGER		Calculated time value in seconds that counts the time elapsed from task capture from first system to source system (for example, fax server to workflow system)
CRT_TIME_FR_SRC_CRTD_SEC	INTEGER	INTEGER		Calculated time value in seconds from time of task creation in the source system (for example, workflow) to the time it was created in iWD.

Attribute	MySQL Type	Oracle Type	PK	Description
ACTIVATE_TIME_FROM_CREATED_SEC	INTEGER	INTEGER		Calculated value in seconds that counts the elapsed time from the time the task was submitted to iWD until activated.
DISTRIBUTE_TIME_FR_CREATED_SEC	INTEGER	INTEGER		Calculated value, in seconds, that counts the elapsed time the task was in iWD (created through to queued) prior to being distributed.
ASSIGN_TIME_FROM_CREATED_SEC	INTEGER	INTEGER		Calculated value, in seconds, that counts the elapsed time from when the task was created in iWD until it was assigned to an agent.
COMPLETE_TIME_FROM_CREATED_SEC	INTEGER	INTEGER		Calculated value, in seconds, counting the elapsed time from when the task was created in iWD until it was completed by the agent.
TOTAL_HELD_TIME_SEC	INTEGER	INTEGER		A calculated value, in seconds, that counts the total time a task was held in iWD. This is a iWD "hold" action via web service or through iWD manager user interface and not a hold event from a softphone or desktop application.
TOTAL_WORK_TIME_SEC	INTEGER	INTEGER		A calculated value, in seconds that counts the time from assigned to an agent until the task was completed by the agent. A task may have multiple work times as noted in task_work_fact. This is the total sum.
CREATED_INTERVAL	INTEGER	INTEGER		Time interval CREATED_DATE_KEY/ CREATED_TIME_KEY. Used for ETL scripts.
COMPLETED_INTERVAL	INTEGER	INTEGER		Time interval COMPLETED_DATE_KEY/ COMPLETED_TIME_KEY. Used for ETL scripts.
DUE_TS	INTEGER	INTEGER		A timestamp for iWD task due date/time. Used for ETL scripts.
COMPLETED_TS	INTEGER	INTEGER		A timestamp for iWD completed date/time. Used for ETL scripts.
ACTIVATION_INTERVAL	INTEGER	INTEGER		Time interval ACTIVATION_DATE_KEY/ ACTIVATION_TIME_KEY. Used for ETL scripts.
ASSIGNED_INTERVAL	INTEGER	INTEGER		Time interval ASSIGN_DATE_KEY/ ASSIGN_TIME_KEY. Used for ETL scripts.
DISTRIBUTED_INTERVAL	INTEGER	INTEGER		Time interval DISTRIBUTED_DATE_KEY/ DISTRIBUTED_TIME_KEY. Used for ETL scripts.
DUE_INTERVAL	INTEGER	INTEGER		Time interval DUE_DATE_KEY/ DUE_TIME_KEY. Used for ETL scripts.
SOURCE_CREATED_INTERVAL	INTEGER	INTEGER		Time interval SOURCE_CREATED_DATE_KEY/ CREATED_TIME_KEY. Used for ETL scripts.
SOURCE_FIRST_CREATED_INTERVAL	INTEGER	INTEGER		Time interval SOURCE_FIRST_CREATED_DATE_KEY/ SOURCE_FIRST_CREATED_TIME_KEY. Used for ETL scripts.
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER		Key to the etl_audit dimension storing attributes related to the ETL job that created the task.
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER		Key to the etl_audit that updated the task fact.
REQUESTED_SKILL_KEY	INTEGER	INTEGER		Agent SKILL that is requested by iWD rule. Key to SKILL dimension.
REQUESTED_AGENT_KEY	INTEGER	INTEGER		Agent that is requested by iWD rule. Key to AGENT dimension.

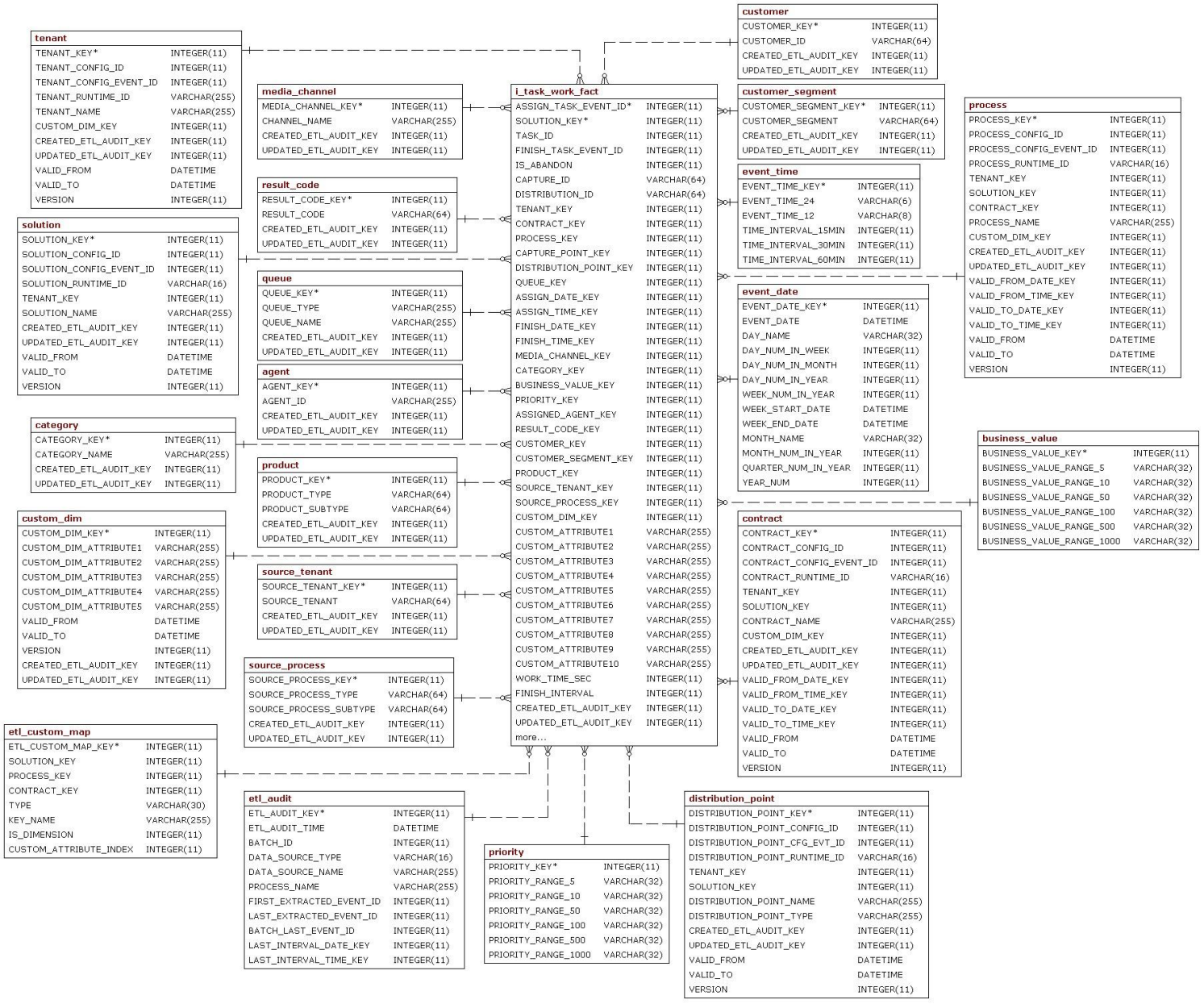
Task Work Fact (task_work_fact)

Task Work Fact describes the task from the perspective of enterprise resources. Each time that a task is assigned to an agent, a new Task Work Fact is recorded. For those installations in which Genesys Info Mart is used, the iWD Data Mart task_work_fact is similar to Interaction Segment Fact in Genesys Info Mart.

In iWD Data Mart database, Task Work Fact can be accessed via the following:

- i_task_work_fact - Intraday data table
- h_task_work_fact - Historical data table
- task_work_fact - Blended (historical and intraday) data view

The following is a view of the task_work_fact star schema, with the standard dimensions surrounding the fact:



The Task Work Fact table contains the following fields:

Attribute	MySQL type	Oracle type	PK	Description
ASSIGN_TASK_EVENT_ID	INTEGER	INTEGER	X	Unique ID returned by the distribution system for the event. In the case of Genesys, this would be the interaction event ID.
SOLUTION_KEY	INTEGER	INTEGER	X	Key to the solution dimension.
TASK_ID	INTEGER	INTEGER		Unique ID for the task.
FINISH_TASK_EVENT_ID	INTEGER	INTEGER		A task can have more than one assigned event ID. This is the ID of the last assigned event.

Attribute	MySQL type	Oracle type	PK	Description
IS_ABANDON	INTEGER	INTEGER		Indicates whether a task has been abandoned (instead of finished normally): • 0 indicates the task was not abandoned (status finished) and • 1 indicates abandon.
CAPTURE_ID	VARCHAR(64)	VARCHAR2(64)		Capture ID for the task, assigned by the source system.
DISTRIBUTION_ID	VARCHAR(64)	VARCHAR2(64)		Distribution ID for the task.
TENANT_KEY	INTEGER	INTEGER		Key to the tenant dimension describing the task's parent iWD tenant.
CONTRACT_KEY	INTEGER	INTEGER		Key to the contract dimension describing the task's parent iWD contract.
PROCESS_KEY	INTEGER	INTEGER		Key to the process dimension describing the task's parent iWD process.
CAPTURE_POINT_KEY	INTEGER	INTEGER		Key to the capture point dimension describing the task's parent iWD capture point (for example, capture point name = XML File capture).
DISTRIBUTION_POINT_KEY	INTEGER	INTEGER		Key to the distribution point dimension describing the task's parent iWD distribution point (for example, name = Toronto Call Center).
QUEUE_KEY	INTEGER	INTEGER		Key to the queue dimension.
ASSIGN_DATE_KEY	INTEGER	INTEGER		Key to the event date for when the task was assigned to agent.
ASSIGN_TIME_KEY	INTEGER	INTEGER		Key to the event time for when the task was assigned to agent.
FINISH_DATE_KEY	INTEGER	INTEGER		Key to the event date for when the task was finished by the agent.
FINISH_TIME_KEY	INTEGER	INTEGER		Key to the event time for when the task was finished by the agent.
MEDIA_CHANNEL_KEY	INTEGER	INTEGER		Key to the media channel dimension describing the media channel for the task (for example, Fax).
CATEGORY_KEY	INTEGER	INTEGER		Key to the business value dimension describing the category associated with the Task.
BUSINESS_VALUE_KEY	INTEGER	INTEGER		Key to the business value dimension.
PRIORITY_KEY	INTEGER	INTEGER		Key to the priority dimension.
ASSIGNED_AGENT_KEY	INTEGER	INTEGER		Key to the agent dimension storing the agent ID for the agent who received the task. This key can be used to retrieve further agent information from Genesys Info Mart, such as Agent Skill, or other employee data from EDW.
RESULT_CODE_KEY	INTEGER	INTEGER		Key to the result code provided by the agent via soft phone.
CUSTOMER_KEY	INTEGER	INTEGER		Key to the customer dimension, storing the unique value to identify the customer. Can be used to retrieve further details about the customer from other enterprise data repositories.
CUSTOMER_SEGMENT_KEY	INTEGER	INTEGER		Key to the customer segment dimension describing the segment for the customer (for example, gold, silver, or bronze).
PRODUCT_KEY	INTEGER	INTEGER		Key to the product dimension, describing the product type (Internet) and sub-type (DSL) associated with the task.
SOURCE_TENANT_KEY	INTEGER	INTEGER		Key to the source_tenant dimension describing the source tenant where the source system is part of a multi-tenant environment.
SOURCE_PROCESS_KEY	INTEGER	INTEGER		Key to the source_process dimension describing the source process (for example, Order)
CUSTOM_DIM_KEY	INTEGER	INTEGER		Key to the custom_dim dimension. This dimension is reserved for future use.
CUSTOM_ATTRIBUTE1	VARCHAR(255)	VARCHAR2(255)		Custom attributes describe a task. A total of 10 customer attributes can be mapped to the task, with 5 additional in the custom_dim dimension.
CUSTOM_ATTRIBUTE2	VARCHAR(255)	VARCHAR2(255)		

Attribute	MySQL type	Oracle type	PK	Description
CUSTOM_ATTRIBUTE3	VARCHAR(255)	VARCHAR2(255)		If more than 10 task attributes exist, the first 10 are mapped with the remaining left unmapped.
CUSTOM_ATTRIBUTE4	VARCHAR(255)	VARCHAR2(255)		
CUSTOM_ATTRIBUTE5	VARCHAR(255)	VARCHAR2(255)		
CUSTOM_ATTRIBUTE6	VARCHAR(255)	VARCHAR2(255)		
CUSTOM_ATTRIBUTE7	VARCHAR(255)	VARCHAR2(255)		
CUSTOM_ATTRIBUTE8	VARCHAR(255)	VARCHAR2(255)		
CUSTOM_ATTRIBUTE9	VARCHAR(255)	VARCHAR2(255)		
CUSTOM_ATTRIBUTE10	VARCHAR(255)	VARCHAR2(255)		
WORK_TIME_SEC	INTEGER	INTEGER		Calculated value, in seconds, where the total work time is time from agent complete to agent assigned.
FINISH_INTERVAL	INTEGER	INTEGER		Time interval FINISH_DATE_KEY/ FINISH_TIME_KEY. Used for ETL scripts.
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER		Key to the etl audit dimension describing ETL job that created the task work fact.
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER		Key to the etl audit dimension describing the ETL job that last updated the task work fact.

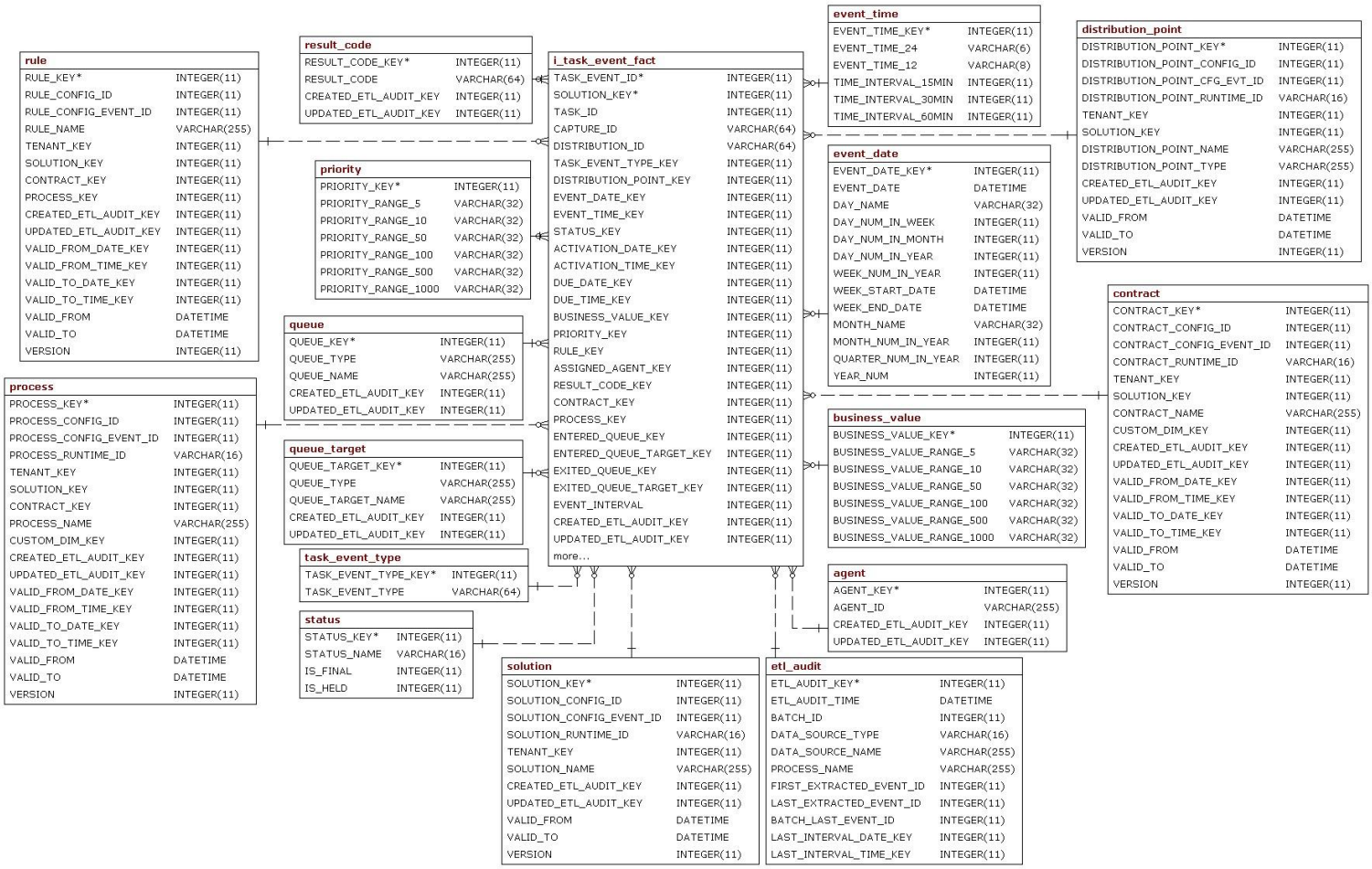
Task Event Fact (task_event_fact)

Task Event Fact provides detailed task audit information. Each change in iWD state creates an audit event in iWD runtime. For example, when a rule is applied, an audit event record records the date and time that the rule was applied, the rule name, and the result of the rule (for example, increase priority from 100 to 110). These events can be seen in iWD Manager runtime view. Each runtime audit event will result in a single Task Event Fact.

In iWD Data Mart database Task Event Fact can be accessed via:

- i_task_event_fact - Intraday data table
- h_task_event_fact - Historical data table
- task_event_fact - Blended (historical and intraday) data view

The following is a view of task_event_fact star schema, with the standard dimensions surrounding the Fact table:



Task Even Fact table with Dimensions

The Task Event Fact table contains the following fields:

Attribute	MySQL Type	Oracle Type	PK	Description
TASK_EVENT_ID	INTEGER	INTEGER	X	Unique ID for the event
SOLUTION_KEY	INTEGER	INTEGER	X	Key to the solution dimension.
TASK_ID	INTEGER	INTEGER		Unique ID for the task.
CAPTURE_ID	VARCHAR(64)	VARCHAR2(64)		Capture ID for the task.
DISTRIBUTION_ID	VARCHAR(64)	VARCHAR2(64)		Distribution ID for the task.
TASK_EVENT_TYPE_KEY	INTEGER	INTEGER		Key to the task event type dimension. Event types are iWD event types like Rule Applied. Event Types can be viewed in the task history in iWD Manager.
DISTRIBUTION_POINT_KEY	INTEGER	INTEGER		Key to the distribution point dimension describing the task's parent iWD distribution point (for example, name = Toronto Call Center).
EVENT_DATE_KEY	INTEGER	INTEGER		Key to the event date for when the event occurred.
EVENT_TIME_KEY	INTEGER	INTEGER		Key to the event time for when the event occurred.
STATUS_KEY	INTEGER	INTEGER		Key to the status dimension.

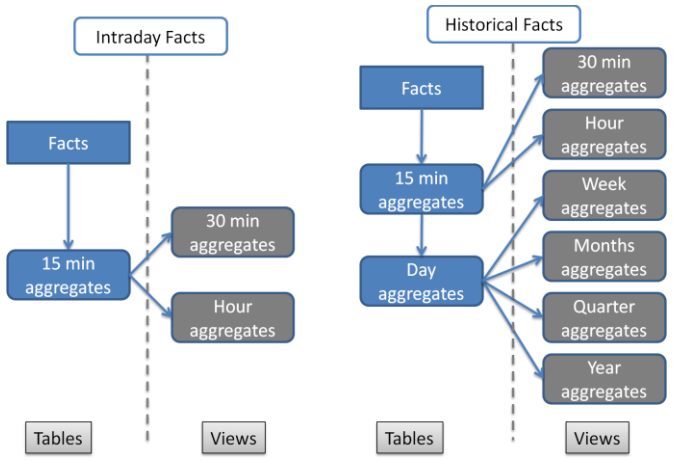
Attribute	MySQL Type	Oracle Type	PK	Description
ACTIVATION_DATE_KEY	INTEGER	INTEGER		Key to the event date dimension describing the iWD task activation date. Tasks can be submitted and not acted upon until this date.
ACTIVATION_TIME_KEY	INTEGER	INTEGER		Key to the iWD task activation time.
DUE_DATE_KEY	INTEGER	INTEGER		Key to the event date dimension describing the task due date as set by iWD rules or by the source system.
DUE_TIME_KEY	INTEGER	INTEGER		Key to the event time dimension describing task due time.
BUSINESS_VALUE_KEY	INTEGER	INTEGER		Key to the business value dimension.
PRIORITY_KEY	INTEGER	INTEGER		Key to the priority dimension.
RULE_KEY	INTEGER	INTEGER		Key to the rule dimension describing the rule that generated the event record. May not apply to all Events and therefore may not be populated for all records in this fact table.
ASSIGNED_AGENT_KEY	INTEGER	INTEGER		Key to the agent dimension for when event record resulted in a task assignment to an agent.
RESULT_CODE_KEY	INTEGER	INTEGER		Key to the result code dimension when the event is an update of a result code by the agent or source system.
CONTRACT_KEY	INTEGER	INTEGER		Key to the contract dimension describing the iWD contract.
PROCESS_KEY	INTEGER	INTEGER		Key to the process dimension describing the iWD process.
ENTERED_QUEUE_KEY	INTEGER	INTEGER		Key to the queue dimension describing the queue in which the task has entered (for event types DISTRIBUTE_WORKBIN and DISTRIBUTE_QUEUE).
ENTERED_QUEUE_TARGET_KEY	INTEGER	INTEGER		Key to the queue target dimension describing the agent, agent group, place or place group to which the task is assigned in new (entered) workbin (for event type DISTRIBUTE_WORKBIN).
EXITED_QUEUE_KEY	INTEGER	INTEGER		Key to the queue dimension describing the queue from which the task has exited (for event types DISTRIBUTE_WORKBIN and DISTRIBUTE_QUEUE).
EXITED_QUEUE_TARGET_KEY	INTEGER	INTEGER		Key to the queue target dimension describing the agent, agent group, place or place group to which the task was assigned in previous (exit) workbin (for event type DISTRIBUTE_WORKBIN and DISTRIBUTE_QUEUE).
CUSTOM_DATA_KEY	INTEGER	INTEGER		Key to the custom dim dimension,
EVENT_INTERVAL	INTEGER	INTEGER		Time interval EVENT_DATE_KEY/ EVENT_TIME_KEY. Used for ETL scripts.
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER		Key to the etl audit dimension storing attributes related to the ETL job that created the task.
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER		Key to the etl audit that updated the task fact.
WORK_TIME_SEC	INTEGER	INTEGER		Nr of seconds that agent worked on task
HELD_TIME_SEC	INTEGER	INTEGER		Nr of seconds that task was in HELD state

Aggregate Facts

Aggregate facts are aggregated representations of the core facts that were described in previous sections. There are three main purposes for aggregated facts:

- Simplified data queries
- Increased query performance
- Decreased database size (granular core facts don't need to be stored for an extended period of time)

Each aggregated fact in iWD Data Mart is an aggregation of two dimensions, one of which is always a time interval. iWD Data Mart directly aggregates facts for two time intervals:15 minutes (intraday and historical) and day (historical).



In addition, the following aggregation levels are supported via database views: 30-Minute, Hourly, Weekly, Monthly, Quarterly, and Yearly.

Each aggregated fact table is postfixed with a time interval: _15min, _30min, _hour, _day, _week, _month, _quarter, or _year.

Similar to core facts, intraday aggregations are prefixed with i_ and historical aggregations are prefixed with h_. Blended aggregations are available only for 15-minute, 30-minute, and hourly time intervals. The following table lists all possible aggregation tables and views per single aggregation type:

Name	Aggregate Type	Type
i_<aggregation_fact>_15min	Intraday 15-min aggregation	Table
i_<aggregation_fact>_30min	Intraday 30-min aggregation	View
i_<aggregation_fact>_hour	Intraday hour aggregation	View
h_<aggregation_fact>_15min	Historical 15-min aggregation	Table
h_<aggregation_fact>_day	Historical day aggregation	Table
h_<aggregation_fact>_week	Historical week aggregation	View
h_<aggregation_fact>_month	Historical month aggregation	View
h_<aggregation_fact>_quarter	Historical quarter aggregation	View
h_<aggregation_fact>_year	Historical year aggregation	View
<aggregation_fact>_15min	Blended 15-min aggregation	View
<aggregation_fact>_30min	Blended 30-min aggregation	View
<aggregation_fact>_hour	Blended hour aggregation	View

Aggregate Task Capture Fact

The Aggregate Task Capture Fact (task_capt_fact) aggregates various task measures, which are grouped by task process, capture point, and business value. Only the tasks that have been classified (status is at least Queued) are included in the aggregations.

Attribute	MySQL Type	Oracle Type	PK	Description
PROCESS_KEY	INTEGER	INTEGER	X	Key to process dimension table.
CAPTURE_POINT_KEY	INTEGER	INTEGER	X	Key to capture_point dimension table.
CUSTOM_DIM_KEY	INTEGER	INTEGER	X	Key to custom_dim dimension table.
BUSINESS_VALUE_KEY	INTEGER	INTEGER	X	Key to business_value dimension table.
INTERVAL_DATE_KEY	INTEGER	INTEGER	X	Key to event_date dimension, indicating start date of time interval. In combination with INTERVAL_TIME_KEY represents the time aggregation axis.
INTERVAL_TIME_KEY	INTEGER	INTEGER	X	Key to event_time dimension, indicating start time of time interval. Applicable only for sub-day aggregations (15 min, 30 min, hour). In combination with INTERVAL_DATE_KEY represents the time aggregation axis.
INTERVAL_KEY	INTEGER	INTEGER		Technical field used by ETL. $= \frac{INTERVAL\_DATE\_KEY \ * 1440 \ + INTERVAL\_TIME\_KEY}{15}$
NEW_TASK_COUNT	INTEGER	INTEGER		How many New tasks were submitted to iWD during the given time interval. Task is counted only after it has been classified (status is at least Queued).
CMPL_TASK_COUNT	INTEGER	INTEGER		How many tasks were Completed during the given time interval.
COMPLETED_OVERDUE_TASK_COUNT	INTEGER	INTEGER		How many tasks have been Completed during the given time interval that had been Overdue; that is, the SLA for the task expired or the Due date/time was not met.
CMPL_TASK_AVG_WORK_TIME	INTEGER	INTEGER		Average agent work time (from Finished date/time stamp - Assigned date/time stamp for a task, see task history in iWD Manager for examples) in seconds for Completed tasks during the given time interval.
CMPL_TASK_AVG_ASSIGN_TIME	INTEGER	INTEGER		Average time before a task was Assigned, measured in seconds and calculated using the task Assigned - Created date/time stamp for Completed tasks during the given time interval. Measures how long, on average, tasks were in backlog before being assigned to an Agent.
CMPL_TASK_AVG_COMPLETE_TIME	INTEGER	INTEGER		Average time before a task was Completed, measured in seconds and calculated using the task Completed - Created date/time stamp for Completed tasks during the given time interval. Measures how long a task was in backlog, similar to Average Assign time, but this measure also includes work time.
CMPL_TASK_AVG_DISTRIBUTE_TIME	INTEGER	INTEGER		Average distribution time (distributed-created) in seconds for completed tasks during the given time interval. Measures how long tasks were in "iWD Backlog" before being distributed to the routing platform (for example, distributed to Genesys Open Media queue).
CMPL_TASK_AVG_SRC_TIME	INTEGER	INTEGER		Average source system time (created-source created) in seconds for completed tasks during the given time interval. Measures the average time a task spent in preceding system before being submitted to iWD. Source Create Date/time is an extended attribute that must be provided by the source system
CMPL_TASK_AVG_PRE_SRC_TIME	INTEGER	INTEGER		Average pre-source system time (source created-first created) in seconds for completed tasks during the given time interval. Same as previous, except beginning time stamp is the system before the preceding system to iWD (for example, fax server to workflow to iWD, fax server is the source first created).
TOTAL_PENDING_TASK_COUNT	INTEGER	INTEGER		Current number of pending (status is Queued, Distributed, Assigned or Held) tasks at the end of the given time interval.
TOTAL_OVERDUE_TASK_COUNT	INTEGER	INTEGER		Current number of pending tasks that areoverdue tasks at the end of the given time interval. A task is considered overdue when the SLA due date/time has been missed.

Aggregate Task Distribution Fact

Aggregate Task Distribution Fact (task_dist_fact) aggregates various task measures, which are grouped by task process, distribution point, and priority over the different time intervals. Only the tasks that have been classified (status is at least Queued) are included in the aggregations.

Attribute	MySQL Type	Oracle Type	PK	Description
PROCESS_KEY	INTEGER	INTEGER	X	Key to process dimension table.
DISTRIBUTION_POINT_KEY	INTEGER	INTEGER	X	Key to distribution_point dimension table.
CUSTOM_DIM_KEY	INTEGER	INTEGER	X	Key to custom_dim dimension table.
PRIORITY_KEY	INTEGER	INTEGER	X	Key to priority dimension table.
INTERVAL_DATE_KEY	INTEGER	INTEGER	X	Key to event_date dimension, indicating start date of time interval. In combination with INTERVAL_TIME_KEY represents the time aggregation axis.
INTERVAL_TIME_KEY	INTEGER	INTEGER	X	Key to event_time dimension, indicating start time of time interval. Applicable only for sub-day aggregations (15 min, 30 min, hour). In combination with INTERVAL_DATE_KEY represents the time aggregation axis.
INTERVAL_KEY	INTEGER	INTEGER		Technical field used by ETL. $= \frac{INTERVAL\_DATE\_KEY * 1440 + INTERVAL\_TIME\_KEY}{15}$
NEW_TASK_COUNT	INTEGER	INTEGER		How many New tasks were submitted to iWD during the given time interval. Task is counted only after it has been classified (status is at least Queued).
CMPL_TASK_COUNT	INTEGER	INTEGER		How many tasks has been completed during the given time interval.
COMPLETED_OVERDUE_TASK_COUNT	INTEGER	INTEGER		How many tasks has been Completed during the given time interval that had been overdue.
CMPL_TASK_AVG_WORK_TIME	INTEGER	INTEGER		Average agent work time (finished - assigned) in seconds for completed tasks during the given time interval.
CMPL_TASK_AVG_ASSIGN_TIME	INTEGER	INTEGER		Average assign time (assigned-created) in seconds for completed tasks during the given time interval.
CMPL_TASK_AVG_COMPLETE_TIME	INTEGER	INTEGER		Average distribution time (distributed-created) in seconds for completed tasks during the given time interval.
CMPL_TASK_AVG_DISTRIBUTE_TIME	INTEGER	INTEGER		Average distribution time (distributed-created) in seconds for completed tasks during the given time interval.
CMPL_TASK_AVG_SRC_TIME	INTEGER	INTEGER		Average source system time (created-source created) in seconds for completed tasks during the given time interval.
CMPL_TASK_AVG_PRE_SRC_TIME	INTEGER	INTEGER		Average pre-source system time (source created-first created) in seconds for completed tasks during the given time interval.
TOTAL_PENDING_TASK_COUNT	INTEGER	INTEGER		Current number of pending (status is Queued, Distributed, Assigned, or Held) tasks at the end of the given time interval.
TOTAL_OVERDUE_TASK_COUNT	INTEGER	INTEGER		Current number of pending tasks that are overdue tasks at the end of the given time interval. A task is considered overdue when the SLA due date/time has been missed.
TOTAL_DISTRIBUTED_TASK_COUNT	INTEGER	INTEGER		Current number of pending tasks that are distributed (status is Distributed or Assigned).

Aggregate Task Classification Fact

Aggregate Task Classification Fact (task_classif_fact) aggregates various task measures, which are grouped by task process, media channel, category, customer segment, media, product, source process, and source tenant over the different time

intervals. Only the tasks that have been classified (status is at least Queued) are included in the aggregations.

Attribute	MySQL Type	Oracle Type	PK	Description
PROCESS_KEY	INTEGER	INTEGER	X	Key to process dimension table.
MEDIA_CHANNEL_KEY	INTEGER	INTEGER	X	Key to media_channel dimension table.
CATEGORY_KEY	INTEGER	INTEGER	X	Key to category dimension table.
CUSTOMER_SEGMENT_KEY	INTEGER	INTEGER	X	Key to customer_segment dimension table.
PRODUCT_KEY	INTEGER	INTEGER	X	Key to product dimension table.
SOURCE_PROCESS_KEY	INTEGER	INTEGER	X	Key to source_process dimension table.
SOURCE_TENANT_KEY	INTEGER	INTEGER	X	Key to source_tenant dimension table.
CUSTOM_DIM_KEY	INTEGER	INTEGER	X	Key to custom_dim dimension table.
INTERVAL_DATE_KEY	INTEGER	INTEGER	X	Key to event_date dimension, indicating start date of time interval. In combination with INTERVAL_TIME_KEY represents the time aggregation axis.
INTERVAL_TIME_KEY	INTEGER	INTEGER	X	Key to event_time dimension, indicating start time of time interval. Applicable only for sub-day aggregations (15 min, 30 min, hour). In combination with INTERVAL_DATE_KEY represents the time aggregation axis.
INTERVAL_KEY	INTEGER	INTEGER		Technical field used by ETL. $= \frac{INTERVAL\_DATE\_KEY \ * \ 1440 \ + INTERVAL\_TIME\_KEY}{15}$
NEW_TASK_COUNT	INTEGER	INTEGER		How many New tasks were submitted to iWD during the given time interval. Task is counted only after it has been classified (status is at least Queued).
CMPL_TASK_COUNT	INTEGER	INTEGER		How many tasks has been completed during the given time interval.
COMPLETED_OVERDUE_TASK_COUNT	INTEGER	INTEGER		How many tasks has been completed during the given time interval that had been overdue.
CMPL_TASK_AVG_WORK_TIME	INTEGER	INTEGER		Average agent work time (finished-assigned) in seconds for completed tasks during the given time interval.
CMPL_TASK_AVG_ASSIGN_TIME	INTEGER	INTEGER		Average assign time (assigned-created) in seconds for completed tasks during the given time interval.
CMPL_TASK_AVG_COMPLETE_TIME	INTEGER	INTEGER		Average distribution time (distributed-created) in seconds for completed tasks during the given time interval.
CMPL_TASK_AVG_DISTRIBUTE_TIME	INTEGER	INTEGER		Average distribution time (distributed-created) in seconds for completed tasks during the given time interval.
CMPL_TASK_AVG_SRC_TIME	INTEGER	INTEGER		Average source system time (created-source created) in seconds for completed tasks during the given time interval.
CMPL_TASK_AVG_PRE_SRC_TIME	INTEGER	INTEGER		Average pre-source system time (source created-first created) in seconds for completed tasks during the given time interval.
TOTAL_PENDING_TASK_COUNT	INTEGER	INTEGER		Current number of pending (status is Queued, Distributed, Assigned, or Held) tasks at the end of the given time interval.
TOTAL_OVERDUE_TASK_COUNT	INTEGER	INTEGER		Current number of pending tasks that are overdue tasks at the end of the given time interval. A task is considered overdue when the SLA due date/time has been missed.

Aggregate Task Agent Fact

Aggregate Task Agent fact (task_agent_fact) aggregates various task measures, which are grouped by task process, assigned agent, and result code over the different time intervals. This aggregate is based on Task Work Fact; so that the numbers are based on how many times an agent has worked on tasks, which could be different from how many tasks are actually completed.

Attribute	MySQL Type	Oracle Type	PK	Description
PROCESS_KEY	INTEGER	INTEGER	X	Key to process dimension table.
AGENT_KEY	INTEGER	INTEGER	X	Key to agent dimension table.
QUEUE_KEY	INTEGER	INTEGER	X	Key to the queue dimension
RESULT_CODE_KEY	INTEGER	INTEGER	X	Key to result_code dimension table.
INTERVAL_DATE_KEY	INTEGER	INTEGER	X	Key to event_date dimension, indicating start date of time interval.
INTERVAL_TIME_KEY	INTEGER	INTEGER	X	Key to event_time dimension, indicating start time of time interval. Applicable only for sub-day aggregations (15 min, 30 min, hour).
INTERVAL_KEY	INTEGER	INTEGER		Technical field used by ETL. $= \frac{INTERVAL\_DATE\_KEY \ * \ 1440 \ + INTERVAL\_TIME\_KEY}{15}$
TASK_WORK_COUNT	INTEGER	INTEGER		How many tasks agent has handled during the given time interval.
AVG_WORK_TIME	INTEGER	INTEGER		Average agent work time on a task (finished-assigned) in seconds.
MIN_WORK_TIME	INTEGER	INTEGER		Minimal agent work time on a task in seconds.
MAX_WORK_TIME	INTEGER	INTEGER		Maximal agent work time on a task in seconds.

Aggregate Task Age Fact

Aggregate Task Age Fact (task_age_fact) aggregates various task age measures, which are grouped by task process and age over the different time intervals. Only the tasks that have been classified (status is at least Queued) are included in the aggregations.

Attribute	MySQL Type	Oracle Type	PK	Description
PROCESS_KEY	INTEGER	INTEGER	X	Key to process dimension table.
AGE_KEY	INTEGER	INTEGER	X	Key to age dimension table (equals number of minutes).
AGE_TYPE	INTEGER	INTEGER	X	Indicates age type: • 1 - since source first created. • 2 - since source created. • 3 - since created. • 4 - since activated. • 5 - since distributed. • 6 - since assigned.
INTERVAL_DATE_KEY	INTEGER	INTEGER	X	Key to the interval date dimension.
INTERVAL_TIME_KEY	INTEGER	INTEGER	X	Key to the interval time dimension.
TOTAL_PENDING_TASK_COUNT	INTEGER	INTEGER		Current number of pending (status is Queued, Distributed, Assigned or Held) tasks at the end of the given time interval.
TOTAL_OVERDUE_TASK_COUNT	INTEGER	INTEGER		Current number of pending tasks that are overdue tasks at the end of the given time interval. A task is considered overdue when the SLA due date/time has been missed.

Aggregate Task Queue Fact

Aggregate Task Queue Fact (task_queue_fact) aggregates various task measures, which are grouped by task distribution queue, process, and priority over the different time intervals. Only the tasks that have been distributed (status is Distributed or Assigned) are included in the aggregations.

Attribute	MySQL Type	Oracle Type	PK	Description
INTERVAL_DATE_KEY	INTEGER	INTEGER	X	Key to the event_date dimension
INTERVAL_TIME_KEY	INTEGER	INTEGER	X	Key to the event_time dimension
QUEUE_KEY	INTEGER	INTEGER	X	Key to queue dimension table.
PROCESS_KEY	INTEGER	INTEGER	X	Key to process dimension table.
CUSTOM_DIM_KEY	INTEGER	INTEGER	X	Key to custom_dim dimension table.
ENTERED_TASK_COUNT	INTEGER	INTEGER		How many tasks have entered (DISTRIBUTE_QUEUE and DISTRIBUTE_WORKBIN iWD task events) the queue during the given time interval.
EXITED_TASK_COUNT	INTEGER	INTEGER		How many tasks have exited the queue (that is, have entered another queue) during the given time interval.
CMPL_TASK_COUNT	INTEGER	INTEGER		How many tasks has been completed during the given time interval.
COMPLETED_OVERDUE_TASK_COUNT	INTEGER	INTEGER		How many tasks has been completed during the given time interval that had been overdue.
CMPL_TASK_AVG_WORK_TIME	INTEGER	INTEGER		Average agent work time (finished-assigned) in seconds for completed tasks during the given time interval.
CMPL_TASK_AVG_ASSIGN_TIME	INTEGER	INTEGER		Average assign time (assigned-created) in seconds for completed tasks during the given time interval.
CMPL_TASK_AVG_COMPLETE_TIME	INTEGER	INTEGER		Average distribution time (distributed-created) in seconds for completed tasks during the given time interval.
CMPL_TASK_AVG_DISTRIBUTE_TIME	INTEGER	INTEGER		Average distribution time (distributed-created) in seconds for completed tasks during the given time interval.
CMPL_TASK_AVG_SRC_TIME	INTEGER	INTEGER		Average source system time (created-source created) in seconds for completed tasks during the given time interval.
CMPL_TASK_AVG_PRE_SRC_TIME	INTEGER	INTEGER		Average pre-source system time (source created-first created) in seconds for completed tasks during the given time interval.
TOTAL_PENDING_TASK_COUNT	INTEGER	INTEGER		Current number of pending (status is Queued, Distributed, Assigned or Held) tasks at the end of the given time interval.
TOTAL_OVERDUE_TASK_COUNT	INTEGER	INTEGER		Current number of pending tasks that are overdue tasks at the end of the given time interval. A task is considered overdue when the SLA due date/time has been missed.

Aggregate Task Queue Age Fact

Aggregate Task Queue Age Fact (task_queue_age_fact) aggregates various task age measures, which are grouped by task distribution queue and age over the different time intervals. Only the tasks that have been classified (status is at least Queued) are included in the aggregations.

Attribute	MySQL Type	Oracle Type	PK	Description
QUEUE_KEY	INTEGER	INTEGER	X	Key to process dimension table.
AGE_KEY	INTEGER	INTEGER	X	Key to age dimension table (equals number of minutes).
AGE_TYPE	INTEGER	INTEGER	X	Indicates age type: • 1 - since source first created. • 2 - since source created. • 3 - since created. • 4 - since activated. • 5 - since distributed. • 6 - since assigned.
INTERVAL_DATE_KEY	INTEGER	INTEGER	X	Key to the interval date dimension
INTERVAL_TIME_KEY	INTEGER	INTEGER	X	Key to the interval time dimension
INTERVAL_KEY	INTEGER	INTEGER		Technical field used by ETL.

Attribute	MySQL Type	Oracle Type	PK	Description
TOTAL_PENDING_TASK_COUNT	INTEGER	INTEGER		Current number of pending (status is Queued, Distributed, Assigned or Held) tasks at the end of the given time interval.
TOTAL_OVERDUE_TASK_COUNT	INTEGER	INTEGER		Current number of pending tasks that are overdue tasks at the end of the given time interval. A task is considered overdue when the SLA due date/time has been missed.
CMPL_TASK_COUNT	INTEGER	INTEGER		How many tasks has been completed during the given time interval.

Dimensions

Dimensions contain static or slowly changing information – as well as information that is used in lookups against the Fact tables - and provide the basis for OLAP/Cube queries. Dimensions in the iWD Data Mart are populated from one of three sources:

- Static values, such as date and time dimensions
- iWD Configuration, as defined in iWD Manager.
- iWD task data

Event Date Dimension

The iWD Data Mart Event Date Dimension is a static dimension that contains date information. The table name is event_date and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
EVENT_DATE_KEY	INTEGER	INTEGER	Primary key
EVENT_DATE	DATETIME	TIMESTAMP	Date of event_key
DAY_NAME	VARCHAR(32)	VARCHAR2(32)	Text name of day (Monday, Tuesday, and so on)
DAY_NUM_IN_WEEK	INTEGER	INTEGER	Day in week (1-7)
DAY_NUM_IN_MONTH	INTEGER	INTEGER	Day in month (ex. 1-31)
DAY_NUM_IN_YEAR	INTEGER	INTEGER	Day in year (1-365)
WEEK_NUM_IN_YEAR	INTEGER	INTEGER	Week number in year (1-52)
WEEK_START_DATE	DATETIME	TIMESTAMP	Date of the first day of the week
WEEK_END_DATE	DATETIME	TIMESTAMP	Date of the last day of the week
MONTH_NAME	VARCHAR(32)	VARCHAR2(32)	Text name of the month (for example, January)
MONTH_NUM_IN_YEAR	INTEGER	INTEGER	Month in year (1-12)
QUARTER_NUM_IN_YEAR	INTEGER	INTEGER	Quarter (1-4)
YEAR_NUM	INTEGER	INTEGER	Year (for example, 2007)

Event Time Dimension

Like event date dimension, the Event Time dimension is a static dimension that contains time information. The table name is event_time and contains the following fields:

Attribute	Oracle Type	Primary	Not Null	Description
EVENT_TIME_KEY	INTEGER	YES	YES	Primary key
EVENT_TIME_24	VARCHAR2(6)	NO	NO	24-hour time, stored in the format HHMMSS; for example, "230159"
EVENT_TIME_12	VARCHAR2(8)	NO	NO	12-hour time, including 'am/pm'; for example, "110159pm"
TIME_INTERVAL_15MIN	INTEGER	NO	NO	The 15-minute interval that the time falls in (1-96)
TIME_INTERVAL_30MIN	INTEGER	NO	NO	The 30-minute interval that the time falls in (1-48)
TIME_INTERVAL_60MIN	INTEGER	NO	NO	The 60-minute interval that the time falls in (1-24)

Tenant Dimension

The Tenant dimension describes the iWD tenant, and the values are populated from the values set up in iWD Manager. As a multi-tenant solution, iWD can be configured with many tenants. The tenant forms part of the ownership chain for a task. The table name is tenant, and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
TENANT_KEY	INTEGER	INTEGER	Primary key
TENANT_CONFIG_ID	INTEGER	INTEGER	iWD Manager ID for the tenant
TENANT_CONFIG_EVENT_ID	INTEGER	INTEGER	Event that created/updated the tenant record
TENANT_RUNTIME_ID	VARCHAR(255)	VARCHAR2(255)	iWD Manager runtime ID for the tenant
TENANT_NAME	VARCHAR(255)	VARCHAR2(255)	Descriptive name of the tenant
CUSTOM_DIM_KEY	INTEGER	INTEGER	Key to custom_dim dimension table containing custom attributes for the tenant.
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that updated this record
VALID_FROM	DATETIME	TIMESTAMP	For the version of the record, the date that it is valid from
VALID_TO	DATETIME	TIMESTAMP	For the version of the record, the date that it is valid to
VERSION	INTEGER	INTEGER	Version of the record

Solution Dimension

Each tenant in iWD can have one or more solutions. A solution can be configured in iWD for testing a new iWD configuration, independent of a production solution. Solution information is populated from iWD Manager configuration. Many fact tables in iWD Data Mart use the solution_key as a primary key.

Attribute	MySQL Type	Oracle Type	Description
SOLUTION_KEY	INTEGER	INTEGER	Primary key
SOLUTION_CONFIG_ID	INTEGER	INTEGER	iWD Manager ID for the solution

Attribute	MySQL Type	Oracle Type	Description
SOLUTION_CONFIG_EVENT_ID	INTEGER	INTEGER	Event that created/updated the solution record
SOLUTION_RUNTIME_ID	VARCHAR(16)	VARCHAR2(16)	iWD Manager runtime ID for the solution
TENANT_KEY	INTEGER	INTEGER	Tenant the solution is associated with
SOLUTION_NAME	VARCHAR(255)	VARCHAR2(255)	Descriptive name of the solution
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that updated this record
VALID_FROM	DATETIME	TIMESTAMP	For the version of the record, the date that it is valid from
VALID_TO	DATETIME	TIMESTAMP	For the version of the record, the date that it is valid to
VERSION	INTEGER	INTEGER	Version of the record

Capture Point Dimension

Tasks are captured through a capture point that is configured in iWD Manager. Capture point information is populated from task information. The table name is capture_point and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
CAPTURE_POINT_KEY	INTEGER	INTEGER	Primary key
CAPTURE_POINT_CONFIG_ID	INTEGER	INTEGER	iWD Manager ID for the capture_point
CAPTURE_POINT_CONFIG_EVENT_ID	INTEGER	INTEGER	Event that created/updated the capture_point record
CAPTURE_POINT_RUNTIME_ID	VARCHAR(16)	VARCHAR2(16)	iWD Manager runtime ID for the capture_point
TENANT_KEY	INTEGER	INTEGER	Tenant the capture_point is associated with
SOLUTION_KEY	INTEGER	INTEGER	Primary key
CAPTURE_POINT_NAME	VARCHAR(255)	VARCHAR2(255)	Descriptive name of the capture_point
CAPTURE_POINT_TYPE	VARCHAR(255)	VARCHAR2(255)	Type of capture point
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that updated this record
VALID_FROM	DATETIME	TIMESTAMP	For the version of the record, the date that it is valid from
VALID_TO	DATETIME	TIMESTAMP	For the version of the record, the date that it is valid to
VERSION	INTEGER	INTEGER	Version of the record

Distribution Point Dimension

Captured tasks are sent for completion via a distribution point. Configured in iWD, the distribution point dimension contains key information about the distribution point. Distribution point is a core iWD attribute. The table name is distribution_point and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
DISTRIBUTION_POINT_KEY	INTEGER	INTEGER	Primary key
DISTRIBUTION_POINT_CONFIG_ID	INTEGER	INTEGER	iWD Manager ID for the distribution_point

Attribute	MySQL Type	Oracle Type	Description
DISTRIBUTION_POINT_CFG_EVT_ID	INTEGER	INTEGER	Event that created/updated the distribution_point record
DISTRIBUTION_POINT_RUNTIME_ID	VARCHAR(16)	VARCHAR2(16)	iWD Manager runtime ID for the distribution_point
TENANT_KEY	INTEGER	INTEGER	Tenant the distribution_point is associated with
SOLUTION_KEY	INTEGER	INTEGER	Solution the capture point is associated with
DISTRIBUTION_POINT_NAME	VARCHAR(255)	VARCHAR2(255)	Descriptive name of the distribution_point
DISTRIBUTION_POINT_TYPE	VARCHAR(255)	VARCHAR2(255)	Type of distribution point
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that updated this record
VALID_FROM	DATETIME	TIMESTAMP	for the version of the record, the date that it is valid from
VALID_TO	DATETIME	TIMESTAMP	for the version of the record, the date that it is valid to
VERSION	INTEGER	INTEGER	version of the record

Contract Dimension

All tasks are associated with a contract that is configured in iWD Manager. The table name is contract and it contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
CONTRACT_KEY	INTEGER	INTEGER	Primary key
CONTRACT_CONFIG_ID	INTEGER	INTEGER	iWD Manager ID for the contract
CONTRACT_CONFIG_EVENT_ID	INTEGER	INTEGER	Event that created/updated the contract record
CONTRACT_RUNTIME_ID	VARCHAR(16)	VARCHAR2(16)	iWD Manager runtime ID for the contract
TENANT_KEY	INTEGER	INTEGER	Tenant the contract is associated with
SOLUTION_KEY	INTEGER	INTEGER	Solution the contract is associated with
CONTRACT_NAME	VARCHAR(255)	VARCHAR2(255)	Descriptive name of the contract
CUSTOM_DIM_KEY	INTEGER	INTEGER	Key to custom_dim dimension table containing custom attributes for the contract.
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that updated this record
VALID_FROM_DATE_KEY	INTEGER	INTEGER	Date key for the contract attribute "valid from"
VALID_FROM_TIME_KEY	INTEGER	INTEGER	Time key for the contract attribute "valid from"
VALID_TO_DATE_KEY	INTEGER	INTEGER	Date key for the contract attribute "valid to"
VALID_TO_TIME_KEY	INTEGER	INTEGER	Time key for the contract attribute "valid to"
VALID_FROM	DATETIME	TIMESTAMP	For the version of the record, the date that it is valid from
VALID_TO	DATETIME	TIMESTAMP	For the version of the record, the date that it is valid to
VERSION	INTEGER	INTEGER	Version of the record

Process Dimension

All tasks are associated with a business process. Processes are associated with a contract in iWD. Populated from iWD Manager configurations, the process dimension contains information on iWD processes. The table name is process and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
PROCESS_KEY	INTEGER	INTEGER	Primary key
PROCESS_CONFIG_ID	INTEGER	INTEGER	iWD Manager ID for the process
PROCESS_CONFIG_EVENT_ID	INTEGER	INTEGER	Event that created/updated the process record
PROCESS_RUNTIME_ID	VARCHAR(16)	VARCHAR2(16)	iWD Manager runtime ID for the process
TENANT_KEY	INTEGER	INTEGER	Tenant the process is associated with
SOLUTION_KEY	INTEGER	INTEGER	Solution the process is associated with
CONTRACT_KEY	INTEGER	INTEGER	Contract the process is associated with
PROCESS_NAME	VARCHAR(255)	VARCHAR2(255)	Descriptive name of the process
CUSTOM_DIM_KEY	INTEGER	INTEGER	Key to custom_dim dimension table containing custom attributes for the process.
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that updated this record
VALID_FROM_DATE_KEY	INTEGER	INTEGER	Date key for the process attribute "valid from"
VALID_FROM_TIME_KEY	INTEGER	INTEGER	Time key for the process attribute "valid from"
VALID_TO_DATE_KEY	INTEGER	INTEGER	Date key for the process attribute "valid to"
VALID_TO_TIME_KEY	INTEGER	INTEGER	Time key for the process attribute "valid to"
VALID_FROM	DATETIME	TIMESTAMP	For the version of the record, the date that it is valid from
VALID_TO	DATETIME	TIMESTAMP	For the version of the record, the date that it is valid to
VERSION	INTEGER	INTEGER	Version of the record

Media Channel Dimension

Media channel describe the type of media, or channel (for example, webform, fax) via which the task was received. It is populated from task information and is a core iWD attribute. The table name is media_channel and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
MEDIA_CHANNEL_KEY	INTEGER	INTEGER	Primary key
CHANNEL_NAME	VARCHAR(255)	VARCHAR2(255)	name of the media channel
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that updated this record

Category Dimensions

Populated from task information, the category dimension contains a list of categories that further describe a task. For example, a category might be a specific type of refund. It is an extended iWD attribute and can be set by the source system. The table name is category and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
CATEGORY_KEY	INTEGER	INTEGER	Primary key
CATEGORY_NAME	VARCHAR(255)	VARCHAR2(255)	Name of the category
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that updated this record

Priority Dimension

Priority is a static dimension. The priority_key represents the value of the priority. The table name is priority and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
PRIORITY_KEY	INTEGER	INTEGER	Primary key
PRIORITY_RANGE_5	VARCHAR(32)	VARCHAR2(32)	Values in the priority granularity of 5; that is 1-5, 6-10 and so on
PRIORITY_RANGE_10	VARCHAR(32)	VARCHAR2(32)	Values in the priority granularity of 10; that is 1-10, 11-20 and so on
PRIORITY_RANGE_50	VARCHAR(32)	VARCHAR2(32)	Values in the priority granularity of 50; that is 1-50, 51-100 and so on
PRIORITY_RANGE_100	VARCHAR(32)	VARCHAR2(32)	Values in the priority granularity of 100; that is 1-100, 101-200 and so on
PRIORITY_RANGE_500	VARCHAR(32)	VARCHAR2(32)	Values in the priority granularity of 500; that is 1-500, 501-1000 and so on
PRIORITY_RANGE_1000	VARCHAR(32)	VARCHAR2(32)	Values in the priority granularity of 1000; that is 1-1000, 1001-2000 and so on

Business Value Dimension

Business value is a static dimension. The business_value_key represents the business value of the task. The table name is business_value and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
BUSINESS_VALUE_KEY	INTEGER	INTEGER	Primary key
BUSINESS_VALUE_RANGE_5	VARCHAR(32)	VARCHAR2(32)	Values in the business value granularity of 5; that is 1-5, 6-10 and so on
BUSINESS_VALUE_RANGE_10	VARCHAR(32)	VARCHAR2(32)	Values in the business value granularity of 10; that is 1-10, 11-20 and so on
BUSINESS_VALUE_RANGE_50	VARCHAR(32)	VARCHAR2(32)	Values in the business value granularity of 50; that is 1-50, 51-100 and so on
BUSINESS_VALUE_RANGE_100	VARCHAR(32)	VARCHAR2(32)	Values in the business value granularity of 100; that is 1-100, 101-200 and so on
BUSINESS_VALUE_RANGE_500	VARCHAR(32)	VARCHAR2(32)	Values in the business value granularity of 500; that is 1-500, 501-1000 and so on
BUSINESS_VALUE_RANGE_1000	VARCHAR(32)	VARCHAR2(32)	Values in the business value granularity of 1000; that is 1-1000, 1001-2000 and so on

Agent Dimension

Populated from task information, the agent dimension contains agents captured from the source system. This is a core iWD attribute. The table name is agent and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description

Attribute	MySQL Type	Oracle Type	Description
AGENT_KEY	INTEGER	INTEGER	Primary key
AGENT_ID	VARCHAR(255)	VARCHAR2(255)	The source system agent ID
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that updated this record

Rule Dimension

Populated from the iWD Manager configuration, the rule dimension contains information on rules. The table name is rule and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
RULE_KEY	INTEGER	INTEGER	Primary key
RULE_CONFIG_ID	INTEGER	INTEGER	iWD Manager ID for the rule
RULE_CONFIG_EVENT_ID	INTEGER	INTEGER	ID to the Event that created/updated the rule record
RULE_NAME	VARCHAR(255)	VARCHAR2(255)	Descriptive name of the rule
TENANT_KEY	INTEGER	INTEGER	Tenant the rule is associated with
SOLUTION_KEY	INTEGER	INTEGER	Solution the rule is associated with
CONTRACT_KEY	INTEGER	INTEGER	Contract the rule is associated with
PROCESS_KEY	INTEGER	INTEGER	Process the rule is associated with
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that updated this record
VALID_FROM_DATE_KEY	INTEGER	INTEGER	Date key for the rule attribute "valid from"
VALID_FROM_TIME_KEY	INTEGER	INTEGER	Time key for the rule attribute "valid from"
VALID_TO_DATE_KEY	INTEGER	INTEGER	Date key for the rule attribute "valid to"
VALID_TO_TIME_KEY	INTEGER	INTEGER	Time key for the rule attribute "valid to"
VALID_FROM	DATETIME	TIMESTAMP	For the version of the record, the date that it is valid from
VALID_TO	DATETIME	TIMESTAMP	For the version of the record, the date that it is valid to
VERSION	INTEGER	INTEGER	Version of the record
RULE_RUNTIME_ID	VARCHAR(255)	VARCHAR2(255)	iWD Manager runtime ID of the rule

Customer Dimension

Tasks are typically associated with a customer. The customer dimension holds the unique identifier - either customer ID or account ID - and is populated from task information. It is an extended iWD attribute. The table name is customer and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
CUSTOMER_KEY	INTEGER	INTEGER	Primary key
CUSTOMER_ID	VARCHAR(64)	VARCHAR2(64)	Unique ID for the Customer - provided by source system
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that updated this record

Customer Segment Dimension

Customer segment is an extended iWD attribute to describe a customer further and can be set by the source system that is submitting the task. A segment oftentimes represents the value of the client to the enterprise – for example, gold, silver or bronze. The table name is customer_segment and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
CUSTOMER_SEGMENT_KEY	INTEGER	INTEGER	Primary key
CUSTOMER_SEGMENT	VARCHAR(64)	VARCHAR2(64)	Name of the customer_segment
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that updated this record

Product Dimension

In some cases, a task may be associated with a specific product - for example, an order for a widget – in which widget is the product. The dimension includes product type and subtype (North American widget versus European widget) and is an extended attribute that can be set by the source system. The table name is product and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
PRODUCT_KEY	INTEGER	INTEGER	Primary key
PRODUCT_TYPE	VARCHAR(64)	VARCHAR2(64)	Type of product
PRODUCT_SUBTYPE	VARCHAR(64)	VARCHAR2(64)	Subtype of product
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that updated this record

Source Process Dimension

Tasks may be associated with a larger process – say, in a workflow system. As an extended attribute, the source process dimension includes a process type and subtype, and can be set by the source system. The table name is source_process and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
SOURCE_PROCESS_KEY	INTEGER	INTEGER	Primary key
SOURCE_PROCESS_TYPE	VARCHAR(64)	VARCHAR2(64)	Name of the source system process, for example: Order
SOURCE_PROCESS_SUBTYPE	VARCHAR(64)	VARCHAR2(64)	Subtype of the process, for example: Activation
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that updated this record

Source Tenant Dimension

In cases in which the source system is part of a multi-tenant system, the source_process and source_product values might not be unique across tenants. The souce tenant dimension provides the ability to define the source of the task further. As

an extended attribute, source tenant can be populated by the source system that is submitting the task. The table name is source_tenant and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
SOURCE_TENANT_KEY	INTEGER	INTEGER	Primary key
SOURCE_TENANT	VARCHAR(64)	VARCHAR2(64)	Name of the tenant from the source system.
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that update this record

Result Code Dimension

The result code dimension is a value that can be set on the completion of a task by the agent. For example, a result code might be set through the agent softphone as a wrap code. The result code dimension is populated from task information and is a iWD extended attribute. The table name is result_code and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
RESULT_CODE_KEY	INTEGER	INTEGER	Primary key
RESULT_CODE	VARCHAR(64)	VARCHAR2(64)	Name of the result_code
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that updated this record

Task Event Type Dimension

The task event type dimension is a static dimension that has the various iWD/Task event types. The table name is task_event_type and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
TASK_EVENT_TYPE_KEY	INTEGER	INTEGER	Primary key
TASK_EVENT_TYPE	VARCHAR(64)	VARCHAR2(64)	The iWD Task event type name; for example, Cancel, Held, Distribute, Assign, Abandon.

Age Dimension

The age dimension is a static dimension that has age ranges to define task age. The table name is age and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
AGE_KEY	INTEGER	INTEGER	Primary key
AGE_MINUTES	INTEGER	INTEGER	Age in minutes at the beginning of interval
AGE_RANGE_15MIN	VARCHAR(64)	VARCHAR2(64)	15-minute range (0-15 minutes, 15 – 30 minutes, …)
AGE_RANGE_1HOUR	VARCHAR(64)	VARCHAR2(64)	1-hour range (0 - 1 hour, 1 – 2 hours, …)
AGE_RANGE_4HOUR	VARCHAR(64)	VARCHAR2(64)	4-hour range (0-4 hours, 4 – 8 hours, …)
AGE_RANGE_8HOUR	VARCHAR(64)	VARCHAR2(64)	8-hour range (0-8 hours, 8 – 16 hours, …)
AGE_RANGE_1DAY	VARCHAR(64)	VARCHAR2(64)	Days (0-1 day, 1–2 days, …)
AGE_RANGE_WEEK	VARCHAR(64)	VARCHAR2(64)	Weeks (0-1 weeks, 1-2weeks, …)

Custom Data Dimension

Each row describes a distinct combination of the custom attributes of a task. Each custom attribute is represented by a string value. The table name is custom_dim and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
CUSTOM_DIM_KEY	INTEGER	INTEGER	Key to customer dimension table
CUSTOM_DIM_ATTRIBUTE1	VARCHAR(255)	VARCHAR2(255)	
CUSTOM_DIM_ATTRIBUTE2	VARCHAR(255)	VARCHAR2(255)	
CUSTOM_DIM_ATTRIBUTE3	VARCHAR(255)	VARCHAR2(255)	
CUSTOM_DIM_ATTRIBUTE4	VARCHAR(128)	VARCHAR2(128)	
CUSTOM_DIM_ATTRIBUTE5	VARCHAR(128)	VARCHAR2(128)	
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	
UPDATED_ETK_AUDIT_KEY	INTEGER	INTEGER	

Metric Dimension

Metrics are a value that can be associated with a contract or process and, optionally, a distribution point. Metrics are set in GLT and are used to associate a value for evaluation against various attributes for reporting purposes (that is, "Cost per Task" set to $0.50). The table name is metric and contains the following fields:

Attribute	MySQL Type	Oracle Type	Description
METRIC_KEY	INTEGER	INTEGER	Primary key.
METRIC_CONFIG_ID	INTEGER	INTEGER	iWD Manager ID for the metric.
METRIC_CONFIG_EVENT_ID	INTEGER	INTEGER	Event that created/updated the metric record.
METRIC_NAME	VARCHAR(255)	VARCHAR2(255)	Name of the metric as set in iWD.
METRIC_DESCRIPTION	VARCHAR(255)	VARCHAR2(255)	Description of what the metric represents as set in iWD.
METRIC_VALUE	VARCHAR(255)	VARCHAR2(255)	Metric value. It is incumbent on the report writer to do any type conversion on this field.
TENANT_KEY	INTEGER	INTEGER	Tenant the metric is associated with.
SOLUTION_KEY	INTEGER	INTEGER	Solution the metric is associated with.
CONTRACT_KEY	INTEGER	INTEGER	Contract the metric is associated with.
PROCESS_KEY	INTEGER	INTEGER	Process the metric is associated with.
DISTRIBUTION_POINT_KEY	INTEGER	INTEGER	Distribution point the metric is associated with.
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that created this record.
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER	etl_audit key that updated this record.
VALID_FROM	DATETIME	TIMESTAMP	For the version of the record, the date that it is valid from.
VALID_TO	DATETIME	TIMESTAMP	For the version of the record, the date that it is valid to.
VERSION	INTEGER	INTEGER	Version of the record.

Queue Dimension

The queue dimension describes queues and workbins in the distribution system. Table name is queue and contains the following fields:

Attribute	MySQL Type	Oracle Type	PK	Description
QUEUE_KEY	INTEGER	INTEGER	X	Primary key.
QUEUE_TYPE	VARCHAR(255)	VARCHAR2(255)		Type of the queue. For Genesys, the following types are supported: • InteractionQueue. • AgentWorkbin. • AgentGroupWorkbin. • PlaceWorkbin. • PlaceGroupWorkbin.
QUEUE_NAME	VARCHAR(255)	VARCHAR2(255)		Name of the queue.
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER		etl_audit key that created this record.
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER		etl_audit key that updated this record.

Queue Target Dimension

The queue target dimension is agent, agent group, place, or place group to which the task is assigned in workbin. Table name is queue_target and contains the following fields:

Attribute	MySQL Type	Oracle Type	PK	Description
QUEUE_TARGET_KEY	INTEGER	INTEGER	X	Primary key.
QUEUE_TYPE	VARCHAR(255)	VARCHAR2(255)		Type of the queue. For Genesys, the following types are supported: • InteractionQueue. • AgentWorkbin. • AgentGroupWorkbin. • PlaceWorkbin. • PlaceGroupWorkbin.
QUEUE_TARGET_NAME	VARCHAR(255)	VARCHAR2(255)		Name of the agent, agent group, place or place group.
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER		etl_audit key that created this record.
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER		etl_audit key that updated this record.

Status Dimension

The status dimension describes iWD task status. The table name is status and contains the following fields:

Attribute	MySQL Type	Oracle Type	PK	Description
STATUS_KEY	INTEGER	INTEGER	X	Primary key
STATUS_NAME	VARCHAR(16)	VARCHAR2(16)		Status name
IS_FINAL	INTEGER	INTEGER		Is set when task status = Completed, Canceled, or Rejected
IS_HELD	INTEGER	INTEGER		Is set when task status = NewHeld or Held
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER		etl_audit key that updated this record

Skill Dimension

Skill dimension describes Genesys agent skills.

Attribute	Mysql Type	Oracle Type	PK	Description
SKILL_KEY	INTEGER	INTEGER	X	Primary key
SKILL_ID	VARCHAR(255)	VARCHAR2(255)		Skill ID
CREATED_ETL_AUDIT_KEY	INTEGER	INTEGER		ETL_AUDIT key that created this record
UPDATED_ETL_AUDIT_KEY	INTEGER	INTEGER		ETL_AUDIT key that updated this record

System Tables

ETL_AUDIT

The ETL_AUDIT table records each executed iWD Data Mart ETL job. It contains the following values:

Attribute	Mysql Type	Oracle Type	PK	Description
ETL_AUDIT_KEY	INTEGER	INTEGER	X	Primary key.
ETL_AUDIT_TIME	DATETIME	TIMESTAMP		Date/time when the job has finished.
BATCH_ID	INTEGER	INTEGER		Batch number, for jobs that processes data in batches (such as Load_Intraday). Each batch will have it's own ETL_AUDIT record.
DATA_SOURCE_TYPE	VARCHAR(16)	VARCHAR2(16)		Primary source of the job data, either "config", "task", or "datamart".
DATA_SOURCE_NAME	VARCHAR(255)	VARCHAR2(255)		Database ID (equals database JDBC URL).
PROCESS_NAME	VARCHAR(255)	VARCHAR2(255)		Process that is processing the records (that is, "load_config_gtl_dm").
FIRST_EXTRACTED_EVENT_ID	INTEGER	INTEGER		The event_id that starts the range of event_ids processed
LAST_EXTRACTED_EVENT_ID	INTEGER	INTEGER		The event_id that ends the range of event_ids processed
BATCH_LAST_EVENT_ID	INTEGER	INTEGER		ID of the last event in the batch. Used to set the starting point for next batch.
LAST_INTERVAL_DATE_KEY	INTEGER	INTEGER		Last date interval processed by the aggregate ETL scripts.
LAST_INTERVAL_TIME_KEY	INTEGER	INTEGER		Last time interval processed by the aggregate ETL scripts.

ETL_CUSTOM_MAP

The ETL_CUSTOM_MAP table is used to configure mapping of task custom attributes to attributes in Task and Task Work fact, as well as the custom data dimension. It contains the following values:

Attribute	MySQL Type	Oracle Type	PK	Description
ETL_CUSTOM_MAP_KEY	INTEGER	INTEGER	X	Unique mapping record key.
SOLUTION_KEY	INTEGER	INTEGER		Key to the solution for which this mapping is valid. If NULL, then applies to all.

Attribute	MySQL Type	Oracle Type	PK	Description
PROCESS_KEY	INTEGER	INTEGER		For which process this mapping is valid, if null then applies to all processes.
CHANNEL_KEY	INTEGER	INTEGER		For which media channel this mapping is valid, if null then applies to all processes.
CATEGORY_KEY	INTEGER	INTEGER		For which category this mapping is valid, if null then applies to all categories.
KEY_NAME	VARCHAR(255)	VARCHAR2(255)		Custom attribute key name in iWD runtime.
IS_DIMENSION	INTEGER	INTEGER		Whether this attribute should be mapped to Custom Data dimension table rather than fact table.
CUSTOM_ATTRIBUTE_INDEX	INTEGER	INTEGER		Custom attribute index, 1-10 if is_dimension is false or 1-5 otherwise.

ETL_KETTLE_LOG

The ETL_KETTLE_LOG table is used to store technical ETL Job information. It contains the following values:

Attribute	MySQL Type	Oracle Type	PK	Description
ID_JOB	INTEGER	INTEGER	X	ID of the ETL Job - assigned by Kettle.
JOBNAME	VARCHAR(50)	VARCHAR2(50)		Name of Job - as entered in Kettle.
STATUS	VARCHAR(15)	VARCHAR2(15)		ETL Job Status; for example, "End".
LINES_READ	INTEGER	INTEGER		Kettle value of lines read.
LINES_WRITTEN	INTEGER	INTEGER		Kettle value of lines written.
LINES_UPDATED	INTEGER	INTEGER		Kettle value of lines updated.
LINES_INPUT	INTEGER	INTEGER		Kettle value of lines input.
LINES_OUTPUT	INTEGER	INTEGER		Kettle value of lines output.
ERRORS	INTEGER	INTEGER		Total number of errors in the Job. See Log_Field for details.
STARTDATE	DATETIME	TIMESTAMP		Job start date.
ENDDATE	DATETIME	TIMESTAMP		End date of the Job.
LOGDATE	DATETIME	TIMESTAMP		Log date.
DEPDATE	DATETIME	TIMESTAMP		
REPLAYDATE	DATETIME	TIMESTAMP		
LOG_FIELD	TEXT	CLOB		Log file for the Job

iWD Data Mart Views

iWD Data Mart provides a number of read-only views of the dimensions, facts and aggregates that are in the iWD Data Mart. iWD Data Mart contains the following views:

- Dimension
- Intraday fact
- Historical fact
- Blended fact (union of intraday and historical facts)
- Intraday aggregate (hour level)
- Historical aggregate (hour, day, week, month, quarter, year levels)
- Blended aggregate (union of intraday and historical hour aggregates)

Views in iWD Data Mart isolate business users and reporting analysts who create reports by using iWD Data Mart data, from changes that might occur in the underlying iWD Data Mart tables.

Chapter 3: ETL Overview

This chapter explains the process by which information is extracted from iWD runtime and configuration databases, transformed, and loaded into the various iWD Data Mart star schemas.

The information in this chapter is organized into the following topics:

What Is ETL?

The data in iWD Data Mart is made available through a process called Extract, Transform and Load - or ETL, for short. The system that is used to create, configure, and execute the ETL process is Kettle, which is part of the Pentaho reporting suite. Kettle top-level objects are Jobs. Jobs are a sequence of steps that are executed according to success/failure criteria. One of the steps that is used by iWD Reporting is transform steps.

iWD Data Mart ETL Jobs

The following are preconfigured ETL jobs that are responsible for creating and populating iWD Data Mart and calculating the various aggregates and measures that are described in Chapter 2 of this guide. The following are the list of ETL Jobs:

Initialize iWD Data Mart

Job Name: GTL_DM_Initialize
Function: Initializes the necessary data structures and populates static dimensions, such as Age Dimension
Schedule: Runs once

Load Configuration

Job Name: GTL_DM_load_config

Function: Loads updates from iWD config tables into dimension tables
Schedule: Configurable through Service Properties; typically runs on a 15-minute cycle, but not more frequently than a 15-minute cycle

Load Intraday

Job Name: GTL_DM_load_intraday
Function: Loads updates from iWD runtime tables into core fact tables
Schedule: Configurable through Service Properties in iWD Manager, but is recommended it be scheduled to run after Load Configuration Job ends through Job Dependency scheduling option.

Aggregate Intraday

Job Name: GTL_DM_aggregate_intraday
Function: Aggregates data that previously was loaded into fact tables by the Load Intraday Job into the Aggregation tables
Schedule: Recommended to schedule immediately after Load Intraday has completed – typically, running every 15 minutes. The frequency of this aggregate job does not have any bearing on the 15-minute aggregate that is being populated.

Aggregate Statistics

Job Name: GTL_DM_aggregate_stats
Function: Generate extended statistics by executing statistics plug-ins
Schedule: Recommended to schedule immediately after Aggregate Intraday since most of the statistics plug-ins are using aggregated facts.

Load Historical

Job Name: GTL_DM_load_historical
Function: Moves data from intraday fact tables into historical fact tables
Schedule: Runs daily through schedule that is defined in Service Properties.

Aggregate Historical

Job Name: GTL_DM_aggregate_historical
Function: Aggregates data from historical fact tables into 15-min, hour, day, and month aggregation tables
Schedule: Runs once a day, after Load Historical

Maintain iWD

Job Name: GTL_DM_maintain
Function: Cleanses historical tables (removes expired facts, based on rules that are defined on the ETL Service property in iWD Manager).
Schedule: Runs once a day, after Aggregate Historical.

Scheduling ETL Jobs

ETL Jobs can be scheduled to run on a recurring basis by using a CRON expression, to run manually, or to run after the successful completion of a dependent service. For more information on the configuration of scheduling ETL jobs, please refer to the *iWD User Guide* and *iWD Deployment and Configuration Guide*.