



Contents

- Understanding the Optimize Schema 1
- Table Naming and Versioning 2

Understanding the Optimize Schema

Optimize stores system and customer data in the “analysis” database within the applicable customer database product (Oracle, SQL Server, etc.). This data forms the schema that defines the structure of a particular database. As delivered, Optimize has a core schema that provides a basis for adding customer-specific metadata and runtime data. As a user creates event maps, facts, and dimensions to work with their specific business operations, Optimize creates the appropriate database tables automatically based on the implemented event maps. These tables form the customer-specific schema that is unique to each Optimize installation.

Because Optimize table names are derived from user-created data structures, it is impossible to provide an accurate, detailed schema diagram for any given user. However, Optimize follows a consistent taxonomy for naming and relating tables, which can be represented by a star diagram. Below is a graphical representation of a basic Optimize star schema:

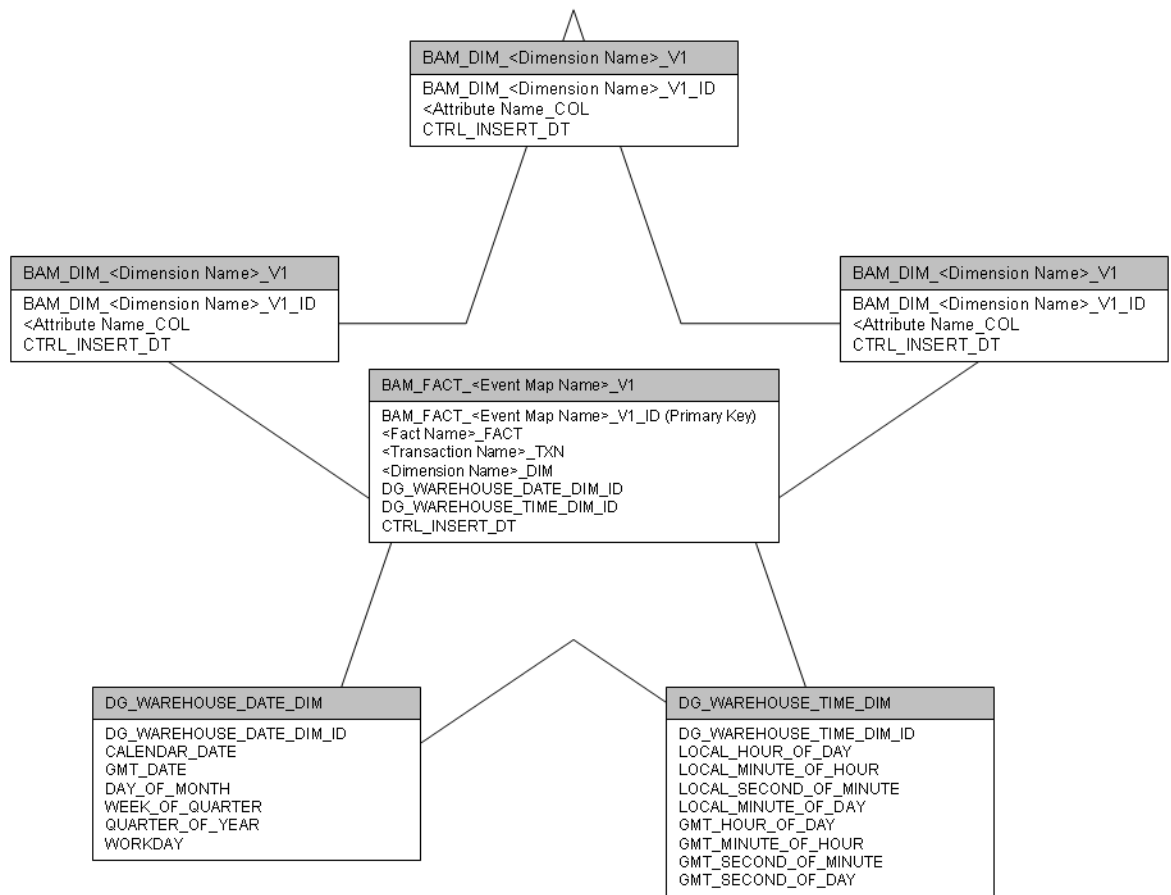


Table Naming and Versioning

Tables and table names in Optimize are derived from the objects they represent, specifically dimensions, facts, and event maps. When you create a dimension, fact, or event map, a new database is created to reflect this new object. There are two types of tables in an Optimize schema: Fact tables and Dimension tables. Note that event maps are metadata, and the tables related to them are beyond the scope of this document.

For example a dimension named Partner would have a corresponding table name of BAM_DIM_PARTNE_V1. If additional Partner dimensions are created, the "V" value is incremented to avoid naming conflicts. To continue our example, if another dimension named PartnershipType were created, the corresponding table would be BAM_DIM_PARTNE_V11. The order of "V" values is V1, V11, V12, V13, V14, and so on.

Fact Tables (BAM_FACT_<Event Map Name>_V1)

Fact tables capture information for every event (that has an event map) generated for Optimize. Each Fact table is associated with one event map and is created after the event map is saved through the Optimize interface in My webMethods. The name of the table is derived from the event map name (not the internal name). All Fact tables contain the following columns:

Column	Description
BAM_FACT_<Event Map Name>_V1_ID	Primary key that uniquely identifies an event row.
<Fact Name>_FACT	Fact columns contain measurable values from events. Usually, though not always, this is a number. There can be any number of fact columns. Examples include order amount, transaction count, and failure percentage.
<Transaction Name>_TXN	Transaction columns contain textual information about an event that is not measurable and not a dimension. Examples include order number, transaction date, and phone number. Optimize does not use this information; it is collected solely for potential use by a reporting tool.
<Dimension Name>_DIM	Dimension columns contain integer values that are foreign keys to dimension tables. Associated dimension tables contain more information related to these dimension foreign keys.
DG_WAREHOUSE_DATE_DIM_ID	This column contains information about the date an event occurred, rather than when it was inserted into the database. The column value is a foreign key reference to the DG_WAREHOUSE_DATE_DIM table, which contains different representations of the date value.
DG_WAREHOUSE_TIME_DIM_ID	This column contains information about the time an event occurred, rather than when it was inserted into the database. The column value is a foreign key reference to the DG_WAREHOUSE_TIME_DIM table, which contains different representations of the time value.
CTRL_INSERT_DT	This is the date and time when an event was inserted into the database. If events from several days ago were delayed and entered today, the CTRL_INSERT_DT value would be today, but the DATE_DIM_ID and TIME_DIM_ID (see above), would be from several days ago. Optimize does not use this field.

Dimension Tables (BAM_DIM_<Dimension Name>_V1)

Dimension tables are linked to Fact tables through a foreign key in the Fact table. The specific column that connects the Fact table to the dimension is named after the Dimension table name. For example, a dimension called Region would have a Dimension table called BAM_DIM_REGION_V1. Any Fact table that uses this dimension would have a column named REGION_DIM that contained a value relating back to the Dimension table.

This document applies to webMethods Optimize Version 8.2 and all previous releases and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 2003-2012 Software AG, Darmstadt, Germany and/or Software AG USA, Inc., Reston, VA, United States of America, and/or their licensors.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at <http://documentation.softwareag.com/legal/>.

Use of this software is subject to adherence to Software AG's licensing conditions and terms. These terms are part of the product documentation, located at <http://documentation.softwareag.com/legal/> and/or in the root installation directory of the licensed product(s).

This software may include portions of third-party products. For third-party copyright notices and license terms, please refer to "License Texts, Copyright Notices and Disclaimers of Third-Party Products". This document is part of the product documentation, located at <http://documentation.softwareag.com/legal/> and/or in the root installation directory of the licensed product(s).