BPM Developer’s Guide for

Semantic V2 History

Randall Kolb

December 2, 2014

# Table of Contents

[Table of Contents 2](#_Toc405218475)

[1 Overview 3](#_Toc405218476)

[2 Data Model 3](#_Toc405218477)

[2.1.1 BPM Semantic 3](#_Toc405218478)

[2.1.2 BPM Semantic V2 3](#_Toc405218479)

[2.1.3 BPM Semantic V2 Data Model Structure 4](#_Toc405218480)

[2.2 Data Model Population Example 5](#_Toc405218481)

[3 Install 11](#_Toc405218482)

[4 Glossary 12](#_Toc405218483)

# Overview

This is a developer’s guide for writing code to populate the Maximus Business Process Management BPM Semantic V2 data models.

# Data Model

Data from the data source is stored in the BPM Semantic or BPM Semantic V2 data models.

### BPM Semantic

The BPM Semantic data model is a star schema dimensional model that contains a record of current and some historical info on the attributes values of the processes monitored. MicroStrategy reports use views of these tables. All older processes and projects use this model. More detailed information and a guide to developing for this model can be found in SVN at svn://rcmxapp1d.maximus.com/maxdat/BPM/doc/BPM\_Developers\_Guide.docx

### BPM Semantic V2

The BPM Semantic V2 History data model (which this document is about) is also a star schema dimensional model that contains a record of current and some historical info on the attributes values of the processes monitored. MicroStrategy reports use views of these tables. The V2 model is different than the original model in that it uses a history table rather than a fact table and uses different views.

### BPM Semantic V2 Data Model Structure

BPM\_D\_DATES is a table containing one row for every day the system stores data for. Often the first BPM\_D\_DATE is set to initialize at one year before first install. For example an install on January 1, 2015 would have BPM\_D\_DATES starting at January 1, 2014 with a new row added every day after install via a daily job that runs a t midnight.

The examples below describe the Manage Work V2 process for MS SQL Server. DDL to create these tables and views can be found in SVN at: svn://rcmxapp1d.maximus.com/maxdat/BPM/Corp/MW\_V2/createdb/mssql/SemanticModel\_MW\_V2.sql

Other processes that use the V2 History model: Process Incidents (MS SQL Server only)

Each process has a data model diagram that can be found here in SVN:

Example: svn://rcmxapp1d.maximus.com/maxdat/BPM/Corp/MW\_V2/doc/mssql/Manage\_Work\_V2\_data\_model\_diagram.pdf

All tables have a view (“\_SV” suffix) . The view is used by the MicroStrategy for reports, never the table directly.

(“SV” stands for semantic view. )

Each process has at least two tables

The current dimension. Contains one row per instance containing the current values of each instance. Some of the columns in the current dimension should be calculated by the ETL when inserting or updating using functions provided for each process. Many of these calculation columns are current date dependent (Example: calculated column AGE\_IN\_BUSINESS\_DAYS uses MV\_V2\_GET\_AGE\_IN\_BUSINESS\_DAYS function) and so are also updated by a job that runs every day at midnight. Only active current (not completed) dimension rows are updated.

Example table: MW\_V2\_CURRENT

Example functions: svn://rcmxapp1d.maximus.com/maxdat/BPM/Corp/MW\_V2/createdb/mssql/MW\_V2\_pkg.sql

The history table. Contains one row per bucket (day) in which an event (change) occurred per instance. The view for the history table (Example view: D\_MW\_V2\_HISTORY\_SV) contains multiple rows per instance – one for every BPM\_D\_DATE entry.

Example table: D\_MW\_V2\_HISTORY

There is another view that identifies the dates that an instance is created, in inventory (active) or completed. This is useful for creating summaries and graphs in MicroStrategy reports of counts for active instances.

Example view: F\_MW\_V2\_BY\_DATE\_SV

Additionally there are usually several dimension table that contain the unique values that are used by several instances. Each history table (Example: D\_MW\_V2\_HISTORY) row has IDs that point to rows in these tables.

Example tables:

D\_BUSINESS\_UNITS

D\_TASK\_TYPES

D\_TEAMS

## Data Model Population Example

MW V2 (Manage Work V2) table name and views are used in the example below.

Values that are in green below must be inserted or updated by ETL.

The views below are then used by MicroStrategy for reports.

{value} below is an example ID pointing to a ID in a dimension table.

Numbers are just made-up examples to show when they change.

Dates below are formatted as:

yyyy-mm-dd hh:mm:ss (MS SQL Server)

YYYY-MM-DD HH24:MI:SS (Oracle)

Each example below is based on one instance (unique MW\_BI\_ID in the where clause of the table or view).

**Event at 2014-11-06 13:00:04 – New Manage Work Instance.**

Insert new **D\_MW\_V2\_HISTORY** table row:

**BUCKET\_START\_DATE  BUCKET\_END\_DATE LAST\_UPDATE\_DATE {value}**

0001-01-01 00:00:00 9999-12-31 00:00:00 2014-11-06 13:00:04 1234561

On day 2014-11-06 this results in:

**D\_MW\_V2\_HISTORY\_SV** view rows: (one row per BPM\_D\_DATE row)

**D\_DATE  LAST\_UPDATE\_DATE {value}**

(earliest BPM\_D\_DATES) 2014-11-06 13:00:04 1234561

… … …

2014-11-06 00:00:00 2014-11-06 13:00:04 1234561

**F\_MW\_V2\_BY\_DATE\_SV** view row

**D\_DATE  CREATION\_COUNT INVENTORY\_COUNT COMPLETION\_COUNT**

2014-11-06 00:00:00 1 1 0

On day 2014-11-07 this results in:

Note: **D\_MW\_V2\_HISTORY** table is unchanged with one row.

**D\_MW\_V2\_HISTORY\_SV** view rows: (one row per BPM\_D\_DATE row)

**D\_DATE  LAST\_UPDATE\_DATE {value}**

(earliest BPM\_D\_DATES) 2014-11-06 13:00:04 1234561

… … …

2014-11-06 00:00:00 2014-11-06 13:00:04 1234561

2014-11-07 00:00:00 2014-11-06 13:00:04 1234561

**F\_MW\_V2\_BY\_DATE\_SV** view rows:

**D\_DATE  CREATION\_COUNT INVENTORY\_COUNT COMPLETION\_COUNT**

2014-11-06 00:00:00 1 1 0

2014-11-07 00:00:00 0 1 0

**Event at 2014-11-08 02:50:34 – Change status**

Update old and insert new **D\_MW\_V2\_HISTORY** table rows:

**BUCKET\_START\_DATE  BUCKET\_END\_DATE LAST\_UPDATE\_DATE {value}**

0001-01-01 00:00:00 2014-11:07 00:00:00 2014-11-06 13:00:04 1234561

2014-11-08 00:00:00 9999-12-31 00:00:00 2014-11-08 02:50:34 1234577

On day 2014-11-08 this results in: (**before** status change at **2014-11-08 02:50:34**)

**D\_MW\_V2\_HISTORY\_SV** view rows: (one row per BPM\_D\_DATE row)

**D\_DATE  LAST\_UPDATE\_DATE {value}**

(earliest BPM\_D\_DATES) 2014-11-06 13:00:04 1234561

… … …

2014-11-08 00:00:00 2014-11-06 13:00:04 1234561

**F\_MW\_V2\_BY\_DATE\_SV** view rows:

**D\_DATE  CREATION\_COUNT INVENTORY\_COUNT COMPLETION\_COUNT**

2014-11-06 00:00:00 1 1 0

2014-11-07 00:00:00 0 1 0

2014-11-08 00:00:00 0 1 0

On day 2014-11-08 this results in: (**after** status change at **2014-11-08 02:50:34**)

**D\_MW\_V2\_HISTORY\_SV** view rows: (one row per BPM\_D\_DATE row)

**D\_DATE  LAST\_UPDATE\_DATE {value}**

(earliest BPM\_D\_DATES) 2014-11-06 13:00:04 1234561

… … …

2014-11-07 00:00:00 2014-11-06 13:00:04 1234561

2014-11-08 00:00:00 2014-11-08 02:50:34 1234577

**F\_MW\_V2\_BY\_DATE\_SV** view rows:

**D\_DATE  CREATION\_COUNT INVENTORY\_COUNT COMPLETION\_COUNT**

2014-11-06 00:00:00 1 1 0

2014-11-07 00:00:00 0 1 0

2014-11-08 00:00:00 0 1 0

**Event at 2014-11-10 12:20:08 – Change status again**

Update old and insert new **D\_MW\_V2\_HISTORY** table rows:

**BUCKET\_START\_DATE  BUCKET\_END\_DATE LAST\_UPDATE\_DATE {value}**

0001-01-01 00:00:00 2014-11:07 00:00:00 2014-11-06 13:00:04 1234561

2014-11-08 00:00:00 2014-11-09 00:00:00 2014-11-08 02:50:34 1234577

2014-11-10 00:00:00 9999-12-31 00:00:00 2014-11-10 12:20:08 1234598

On day 2014-11-10 this results in: (**after** status change at 2014-11-10 12:20:08)

**D\_MW\_V2\_HISTORY\_SV** view rows: (one row per BPM\_D\_DATE row)

**D\_DATE  LAST\_UPDATE\_DATE {value}**

(earliest BPM\_D\_DATES) 2014-11-06 13:00:04 1234561

… … …

2014-11-07 00:00:00 2014-11-06 13:00:04 1234561

2014-11-08 00:00:00 2014-11-08 02:50:34 1234577

2014-11-09 00:00:00 2014-11-08 02:50:34 1234577

2014-11-10 00:00:00 2014-11-10 12:20:08 1234598

**F\_MW\_V2\_BY\_DATE\_SV** view rows:

**D\_DATE  CREATION\_COUNT INVENTORY\_COUNT COMPLETION\_COUNT**

2014-11-06 00:00:00 1 1 0

2014-11-07 00:00:00 0 1 0

2014-11-08 00:00:00 0 1 0

2014-11-09 00:00:00 0 1 0

2014-11-10 00:00:00 0 1 0

**Event at 2014-11-10 14:30:19 – Change status again on the same day.**

Events later in the day (where bucket interval = 1 day) overwrite the previous events during that same day.

Update **D\_MW\_V2\_HISTORY** table row:

**BUCKET\_START\_DATE  BUCKET\_END\_DATE LAST\_UPDATE\_DATE {value}**

0001-01-01 00:00:00 2014-11:07 00:00:00 2014-11-06 13:00:04 1234561

2014-11-08 00:00:00 2014-11-09 00:00:00 2014-11-08 02:50:34 1234577

2014-11-10 00:00:00 9999-12-31 00:00:00 2014-11-10 14:30:19 1234623

On day 2014-11-10 this results in: (**after** status change at 2014-11-10 12:20:08)

**D\_MW\_V2\_HISTORY\_SV** view rows: (one row per BPM\_D\_DATE row)

**D\_DATE  LAST\_UPDATE\_DATE {value}**

(earliest BPM\_D\_DATES) 2014-11-06 13:00:04 1234561

… … …

2014-11-07 00:00:00 2014-11-06 13:00:04 1234561

2014-11-08 00:00:00 2014-11-08 02:50:34 1234577

2014-11-09 00:00:00 2014-11-08 02:50:34 1234577

2014-11-10 00:00:00 2014-11-10 14:30:19 1234623

**F\_MW\_V2\_BY\_DATE\_SV** view rows:

**D\_DATE  CREATION\_COUNT INVENTORY\_COUNT COMPLETION\_COUNT**

2014-11-06 00:00:00 1 1 0

2014-11-07 00:00:00 0 1 0

2014-11-08 00:00:00 0 1 0

2014-11-09 00:00:00 0 1 0

2014-11-10 00:00:00 0 1 0

**Event at 2014-11-12 09:20:22 – Change status and complete.**

Update old and insert new **D\_MW\_V2\_HISTORY** table rows:

**BUCKET\_START\_DATE  BUCKET\_END\_DATE LAST\_UPDATE\_DATE {value}**

0001-01-01 00:00:00 2014-11:07 00:00:00 2014-11-06 13:00:04 1234561

2014-11-08 00:00:00 2014-11-09 00:00:00 2014-11-08 02:50:34 1234577

2014-11-10 00:00:00 2014-11-12 00:00:00 2014-11-10 14:30:19 1234623

2014-11-12 00:00:00 9999-12-31 00:00:00 2014-11-12 09:20:22 1234752

On day 2014-11-12 this results in: (**after** status change at 2014-11-12 09:20:22)

**D\_MW\_V2\_HISTORY\_SV** view rows: (one row per BPM\_D\_DATE row)

**D\_DATE  LAST\_UPDATE\_DATE {value}**

(earliest BPM\_D\_DATES) 2014-11-06 13:00:04 1234561

… … …

2014-11-07 00:00:00 2014-11-06 13:00:04 1234561

2014-11-08 00:00:00 2014-11-08 02:50:34 1234577

2014-11-09 00:00:00 2014-11-08 02:50:34 1234577

2014-11-10 00:00:00 2014-11-10 14:30:19 1234623

2014-11-11 00:00:00 2014-11-10 14:30:19 1234623

2014-11-12 00:00:00 2014-11-12 09:20:22 1234752

**F\_MW\_V2\_BY\_DATE\_SV** view rows:

**D\_DATE  CREATION\_COUNT INVENTORY\_COUNT COMPLETION\_COUNT**

2014-11-06 00:00:00 1 1 0

2014-11-07 00:00:00 0 1 0

2014-11-08 00:00:00 0 1 0

2014-11-09 00:00:00 0 1 0

2014-11-10 00:00:00 0 1 0

2014-11-11 00:00:00 0 1 0

2014-11-12 00:00:00 0 0 1

Since the instance is now no longer active:

The D\_MW\_V2\_HISTORY table and F\_MW\_V2\_BY\_DATE\_SV view remain unchanged forever.

The D\_MW\_V2\_HISTORY\_SV view returns an additional new row every day forever.

# Install

Every project has an install guide that describes the steps and files need to install the MAXdat schema.

Example: svn://rcmxapp1d.maximus.com/maxdat/BPM/UK\_HWS/doc/mssql/DB\_install\_UK\_HWS\_README.txt

# Glossary

**BPM** – Business Process Management

**Instance** – In MAXdat an instance is a path through a business process flow chart, made by a particular client, such as one client applying for benefits. This translates into one row in the current dimension table in the semantic data model that describes the detailed state of that process for the client

**MicroStrategy** – Business intelligence and reporting software.

**Process** – A business process described by a requirement spreadsheet. Example: Manage Work

**Project** – A set of business processes used to fulfill contractual obligations for BPM for an entity such as a state benefit program. Example: UK Health for Work Service (UK-HWS)

**SVN** – Subversion – Software revision control system.