



Report and Dashboard Template 10.1.1

User Guide

Introduction

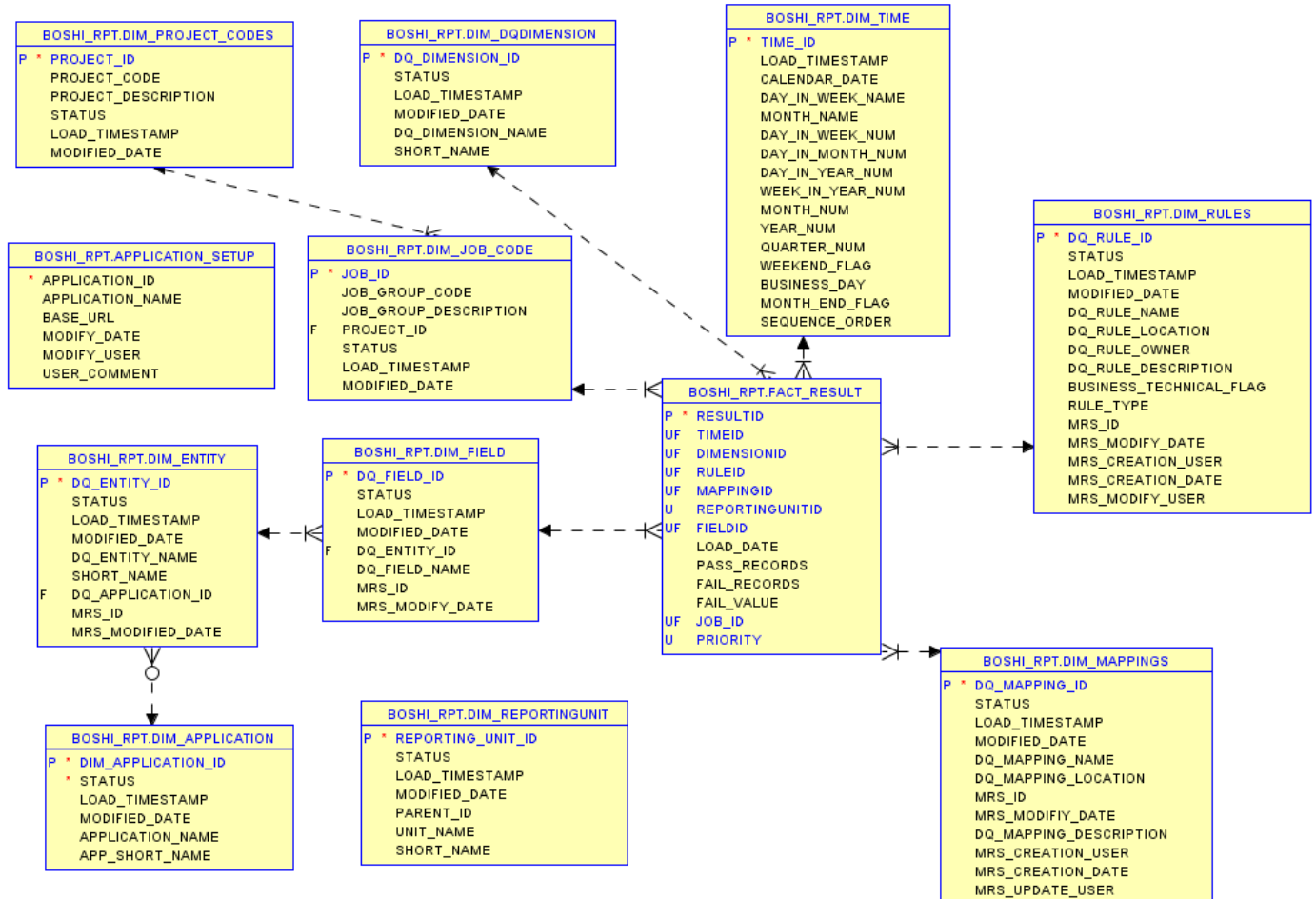
The Informatica Data Quality Reporting and Dashboard Template – for Informatica Data Quality 10.1.1, is designed to provide you a framework to capture reporting metrics for data quality issues. The template includes a database schema, sample data, mappings to load Model Repository Service (MRS) data into dimension tables, sample reports and Dashboards illustrating DQ metrics and User Activity in the MRS. In addition it includes sample data to load into the schema to allow you to view the reports without having the system up and running and generating metrics.



Reporting and Dashboard Technical Overview

Reporting Schema

The following diagram outlines the Reporting Schema used by the template for reporting data quality metrics.



The following is the table definition used to monitor activity in the repository.

BOSHI_RPT.MRS_AUDIT_HISTORY	
P * CID	NUMBER (19)
OID	NUMBER (19)
PROJECT_NAME	VARCHAR2 (765 BYTE)
OBJECT_TYPENAME	VARCHAR2 (765 BYTE)
OBJECT_NAME	VARCHAR2 (255 BYTE)
OBJECT_PATH	VARCHAR2 (3999 BYTE)
CREATED_BY	VARCHAR2 (255 BYTE)
P * CREATION_DATE	DATE
UPDATED_BY	VARCHAR2 (255 BYTE)
P * UPDATED_DATE	DATE
MRS_AUDIT_HISTORY_PK (CID, CREATION_DATE, UPDATED_DATE)	
MRS_AUDIT_HISTORY_PK (CID, CREATION_DATE, UPDATED_DATE)	

The schema utilizes a star schema format which captures aggregated pass/ fail metrics for data quality tests across a number of dimensions.

Table Name	Description
Fact_Result	Base table in the schema for recording DQ metrics.
Fact_Drilldown	Table records key information for each record that fails a DQ test. The table records primary key information of the source record to allow trace back to the source.
DIM_DQDIMENSION	Stores dimension information related to data quality dimension such as Completeness, consistency, conformity, etc. This table is prepopulated with common data Quality dimensions.
DIM_FIELD	Stores information related to specific fields being evaluated. This table has an enforced dependency on the DIM_ENTITY table.
DIM_ENTITY	Container object for all data fields that are part of the entity being tested. For example, a field might refer to a person name, the entity would be the contact. This table has an unenforced dependency on DIM_APPLICATION.
DIM_APPLICATION	Container object encapsulating all entities that are part of an application. For example, Contact, Lead and Account are all parts of an application such as Salesforce.com. This table has an unenforced dependency with the APPLICATION_SETUP table.
APPLICATION_SETUP	Stores details that allow url to be generated to open a web based application on a specific record in conjunction with the FACT_DRILLDOWN and FACT_RESULT tables.
DIM_JOB_CODE	Stores jobs and job codes that may be used to track mappings or applications that are executed as part of a project. This table has an enforced dependency on the DIM_PROJECT_CODES table.
DIM_PROJECT_CODES	Container object encapsulating all jobs that are part of a particular project.
DIM_REPORTINGUNIT	Stores details or any business or reporting units in an organization that are responsible for the quality of data in those businesses units.
DIM_MAPPINGS	Stores details of all mappings stored in the MRS and are run against DQ rules. This table can be populated using mappings, m_load_rule_and_mapping_dimensions and m_load_dim_mappings_from_PWH, in the Dashboard and Reporting template.
DIM_RULES	Stores details of all rules and mapplets stored in the MRS. This table can be populated using mapping, m_load_rule_and_mapping_dimensions, in the Dashboard and

	Reporting template.
DIM_TIME	Stores all calendar dates and various dimensions related to time and dates that user may want to use to analyze their data. Data is provided to populate this table up to Dec 31, 2050
MRS_AUDIT_HISTORY	Standalone table independent of the star schema. This is a denormalized view that stores updates to the MRS. This table is populated by mapping m_load_mrs_audit_history and executed via application: app_mrs_audit_history

The view VIEW_ALL_FACT_RESULTS combines data from most tables and can be used for most common queries against the schema. Tables not include in this view include FACT_DRILLDOWN and MRS_AUDIT_HISTORY.

Data Quality Mappings

The Reporting and Dashboard template supplies a number of mappings to support usage of the template. The mappings are used for both practical and demo purposes.

Mapping Name	Description
m_load_mrs_audit_history	Mapping to load data from the MRS view MRX_OBJECT_SUMMARY into the reporting schema table MRS_AUDIT_HISTORY. The mapping checks to see if the object and update time and new or the same as a previous entry. Any new entries are added to MRS_AUDIT_HISTORY. Entries that already exist are rejected.
m_load_rule_and_mapping_dimensions	Mapping to load data DIM_MAPPINGS and DIM_RULES tables with mapping and rule information from the MRS. If a rule or mapping already exists, information will be updated, if a rule or mapping is new then data is inserted into the above tables.
m_load_dim_mappings_from_PWH	Mapping to load DIM_MAPPINGS with profile that are stored in the MRS. If a profile already exists, information will be updated, if a rule is new, then data is inserted.
m_Evaluate_Address	Demo mapping to demonstrate loading the FACT_RESULT and FACT_DRILLDOWN tables. The mapping utilizes a number of lookups to obtain codes related to the various dimensions and evaluates the data source based on logic contained in the mapping itself.
m_Evaluate_Contact	Demo mapping to demonstrate loading the FACT_RESULT and FACT_DRILLDOWN tables. The mapping utilizes a number of lookups to obtain codes related to the various dimensions and evaluates the data source based on logic contained in the mapping itself.
m_Read_Uniqueness_from_PWH	Demo mapping to demonstrate loading the FACT_RESULT and FACT_DRILLDOWN tables. The mapping utilizes a number of lookups to obtain codes related to the various dimensions. This mapping accesses data stored in the profile warehouse and extracts uniqueness details for all profile and fields stored in there. In cases

where an entity and field contained in the profile match entries in DIM_FIELDS and DIM_ENTITY data will be written to the FACT_RESULT Table. If not, entries are rejected.

Other Data Quality Objects

A number of other objects are provided along with the mappings. This section provides a brief overview of the other objects.

Unmanaged Reference Tables

A number of unmanaged reference tables are provided along that access the dimension tables in the star schema. These tables provide users with an interface to view and edit data in the dimension tables.

The following tables have corresponding reference table entries in the MRS:

DIM_DQDIMENSION
DIM_FIELD
DIM_ENTITY
DIM_APPLICATION
APPLICATION_SETUP
DIM_JOB_CODE
DIM_PROJECT_CODES
DIM_REPORTINGUNIT
DIM_MAPPINGS
DIM_RULES
DIM_TIME

Physical Data Objects

A number of physical data objects are available which link to database various database repositories including:

- Reporting schema used by the template
 - MRS_AUDIT_HISTORY
 - FACT_RESULT
 - FACT_DRILLDOWN
 - DIM_MAPPINGS
 - DIM_RULES
- MRS repository
 - Profile_List



- Mappings_Rules_with_)user_Details
 - MRX_MAPPINGS
 - MRX_OBJECT_SUMMARY
- Profile Warehouse Repository
 - PWH_Extract

Applications

One application, `app_mrs_audit_history`, executes the mapping `m_load_mrs_audit_history`. The application can be executed from the command line using scheduling software to allow you to capture a near-real-time update history of the MRS.

Logical Data Object

A logical data object, `Fact_result_Insert`, is provided to capture and control the aggregation and output of results and bad records into the `FACT_RESULT` and `FACT_DRILLDOWN` tables.

Dependencies from Other Projects

The Reporting and Dashboard template also has dependencies on 2 projects: `Informatica_CRM_Template` and `Informatica_DQ_Content`. These dependencies are provided as part of the installation package for the Reporting and Dashboard template. Depending on what content you have obtained from Informatica, you may need to add these dependencies to your repository or reuse/replace the content in your repository with your existing content or some combination of these 2 options.

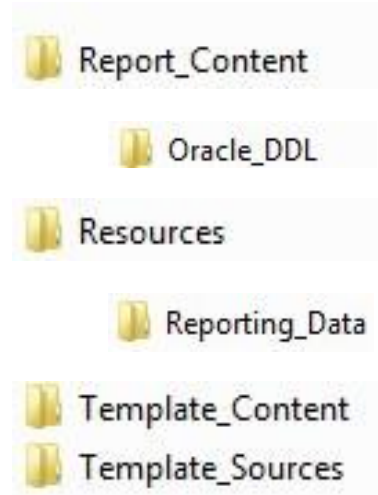
informatica

Installation Steps

Package Name

Informatica_Report_Dashboard_1011.zip

Package Contents



Folder	Description
Report_Content	Contains the DDL scripts used to create the Oracle database tables, views and sequences used in the reporting schema.
Resources	Contains optional data files that can be used by the users to populate the dimension tables. Data for the Fact Result table has also been provided to allow for a quick demonstration of the reports without setting up the test mappings. Data is provided in SQL Insert scripts.
Template_Content	Contains IDQ xml files and Reference data zip files to be installed in the MRS.
Template_Sources	Contains test data files to be copied to the DQ server. Use content installer to install.



Infrastructure

- Informatica Server 10.1.1
- Informatica Developer 10.1.1
- Oracle Database

Service and Database Configuration

You configure services and database connections when you install Informatica Data Quality. Use Informatica Administrator to configure the following services:

- Model Repository Service
- Data Integration Service
- Analyst Service
- Content Management Service

Note: Create the Informatica staging database and profile warehouse database connections before you configure Informatica services.

Installing the Reporting and Dashboard Template

The steps to install and configure the template components affect Informatica Data Quality.

To install a Reporting and Dashboard template, complete the following tasks:

1. Create the reporting schema in the Oracle database.
2. Create the Reporting Dashboard connection.

Step 1: Create the Reporting Schema

1. Open an SQL session and connect to the custom schema in the Oracle database.
 - Note: You can create a custom schema or connect to an existing schema in the Oracle database.
2. Run the following DDL scripts in the Oracle DDL directory:
 1. Dashboard_sequences_v2.sql
 2. Dashboard_tables_All_DDL_v2.sql
 3. VIEW_ALL_FACT_RESULTS.sql

Step 2: Create the Reporting Dashboard Connection

1. In the Administrator tool, click the Domain tab.
2. Click the Connections view.
3. In the Navigator, select the domain.
4. In the Navigator, click Actions > New > Connection. The New Connection dialog box appears.
5. In the New Connection dialog box, select Oracle as the connection type.
6. Enter the database connection properties to connect to the Oracle database where you have created the custom schema.

Note: The connection name for the Reporting Dashboard connection must be unique.



informatica

Verifying the Reporting and Dashboard Template

You can install sample data to verify the operations of the Reporting and Dashboard template. To verify the operations of the Reporting and Dashboard template, complete the following steps:

1. Import XML data and reference data into the Model repository.
2. Install the demonstration source data. Use the Data Quality Content Installer to install the data.
3. Load the dimension tables.
4. Verify mapping operations in Informatica Developer.

Step 1: Import XML and Reference Data into the Model Repository

1. Open the Developer tool and connect to the Model repository.
 2. If they do not already exist, create connections to the MRS, Reporting schema and Profile Warehouse.
 3. Create a project named Informatica_Report_Template. Verify that the project is shared.
 4. Import the XML file from the Template_Content folder to the Informatica_Report_Template project. Select the Import Object Metadata File (Advanced) option in the import wizard.
 5. Select the reference data compressed file in the Template_Content folder when prompted by the import wizard. If the Model repository contains an Informatica_DQ_Content and Informatica_CRM_Template projects, use the dependencies from that project when you import the data. If the Informatica_DQ_Content and Informatica_CRM_Template projects are not present, use the dependencies in the XML file.
- Note: If you use the dependencies in the XML file and an Informatica_DQ_Content and Informatica_CRM_Template projects are present in the repository, you overwrite the configuration of the Informatica_DQ_Content and Informatica_CRM_Template projects.

Step 2: Install Demonstration Source Data

Use the Data Quality Content Installer to install the demonstration data files to the Informatica Data Quality services host machine.

1. Run the Data Quality Content installer executable file.
 - a. On Windows, run Content_Installer_server.exe
 - b. On UNIX, run Content_Installer_server.sh
 - c. The install wizard starts.

Enter the root directory of the Informatica services installation. This may be a remote directory. Browse to this directory if required.
2. Select the demonstration source data compressed file in the Template_Sources folder. When you install the demonstration source data files, enter the path to the directory where the installer writes these files.

Step 3: Load the Dimension Tables

1. In the Reporting and Dashboard installation package, browse to the following directory:
 - Reporting Data/SQL Insert Scripts
2. Open an SQL session.
3. Load and run the following scripts in the Oracle database in which you created the custom schema:
 - a. DIM_DQDIMENSION_DATA_TABLE.sql
 - b. DIM_APPLICATION_DATA_TABLE.sql
 - c. DIM_ENTITY_DATA_TABLE.sql
 - d. DIM_FIELD_DATA_TABLE.sql
 - e. DIM_MAPPINGS_DATA_TABLE.sql
 - f. DIM_PROJECT_CODES_DATA_TABLE.sql
 - g. DIM_JOB_CODE_DATA_TABLE.sql
 - h. DIM_REPORTINGUNIT_DATA_TABLE.sql
 - i. DIM_RULES_DATA_TABLE.sql
 - j. DIM_TIME_DATA_TABLE.sql
3. Run FACT_RESULT.sql and MRS_AUDIT_HISTORY_DATA_TABLE.sql to load the database tables with the result data without running the test mappings.



informatica

Step 4: Verify the Informatica Mappings

1. Open the Developer tool.
2. Open the project that contains the template mappings.
3. Run the mappings in the Address and Contacts folders.
4. Log into the reporting schema you created. Check VIEW_ALL_FACT_RESULTS view to see if data from your mapping run is visible here.