Setup Guide

Multi-fact Table Setup Guide

This **Setup Guide** is designed to assist users with updating an existing i2b2 environment to be able to query multiple fact tables.

Requirements

The following items are required for multi-fact tables to work correctly.

- You have a working version of i2b2 that is running release 1.7.12 or higher.
- All your fact tables reside in the same database.
- All fact tables have the same data structure including primary keys & indexes.
- **observation_fact** remains as one of the fact tables

Assumptions

This setup guide assumes the following:

- You know how to access and update your database using SQL editor
- You are familiar with i2b2 data; metadata & data mart tables. *(See i2b2 Data Mart & Ontology page for a quick overview)*
- Your fact tables have been setup.
- Your fact tables are organized by concept: e.g. all diagnoses facts are in one table; all medication facts in another

Setup Process

The process of setting up the i2b2 to use multiple fact tables is straight forward. Once you have updated your software to at least version 1.7.12 or higher, create your new fact tables and load the data into them. There are only two remaining steps.

1. Update the terms in your metadata tables so the i2b2 server knows what fact table to query
2. Turn the Multi-fact table feature on in the *hive_cell_params* table. Refer to Multi-fact table home

This setup process is designed to assist users with updating an existing i2b2 environment to be able to query multiple fact tables
.

Update Ontology Tables

How classic i2b2 works: The standard i2b2 star schema only allows for one fact table. In the i2b2 database this is the observation_fact
table. When running a query in the i2b2, the CRC queries against the observation_fact table by default.

Update Metadata:

The CRC needs to know the name of the fact table to search when a term is used in a query. This is done by

updating the metadata tables for each term you want redirected to an alternate fact table. You need to

update the value in c_facttablecolumn column to now be facttablename. columnname. Examples are provided below.

The good news is, you only need to update those terms you want to redirect to a different fact table. If the term is still going to the observation_fact table you do not need to update it. If a table name does not preface the column name the system will work as it does now and the observation_fact table will be searched.

Example:concept_cd vs. lab_facts.concept_cd

The sample tables shown below are only displaying 5 of the 25 columns that are currently part of the i2b2 metadata tables.

Single fact table

This sample shows how the i2b2 metadata table is setup to work with a single fact table.

c_name	c_basecode	c_facttablecolumn	c_tablename	c_columnname
Diabetes mellitus	ICD9:250.00	concept_cd	concept_dimension	concept_path
Amoxicillin	NDC: 559553071670	concept_cd	concept_dimension	concept_path
Hematocrit	LOINC:4544-3	concept_cd	concept_dimension	concept_path
X-Ray Report	LCS-i2b2: XR_RPT	concept_cd	concept_dimension	concept_path

Multiple fact tables

In this sample, we take the above table and update it to point to multiple fact tables. The diagnoses are still going to the observation_fact table, the other three facts are going to one of these three new fact tables:

1. medication_fact (Amoxicillin)

2. lab_fact (Hematocrit)

3. diagnostic_fact (X-Ray Report)

Using this example, the i2b2 metadata table would look like:

c_name	c_basecode	c_facttablecolumn	c_tablename	c_columnname
Diabetes mellitus	ICD9:250.00	concept_cd	concept_dimension	concept_path
Amoxicillin	NDC: 559553071670	medication_fact. concept_cd	concept_dimension	concept_path
Hematocrit	LOINC:4544-3	lab_fact.concept_cd	concept_dimension	concept_path
X-Ray Report	LCS-i2b2: XR_RPT	diagnostic_fact. concept_cd	concept_dimension	concept_path

In the above example, you will notice the entry for Diabetes does not have the prefix 'observation_fact.'. This is because it is not needed. If a prefix does not exist the system will automatically look to the observation_fact table

Update *multifacttable* parameter

Using SQL editor, update the param_name_cd *queryprocessor.multifacttable value to true* in *hive_cell_params* table. When the value is set to true the CRC will now look for multiple fact tables. By default, this value is set to false. If you are going to be using multiple fact tables you will need to update this value to be true.

Example in Oracle, use the following update statement to manually update the value to true:

UPDATE h*ive_cell_params* set value ='true' where param_name_cd ='queryprocessor.multifacttable' and cell_id ='CRC';