



Project Management Design Specification

Status

Document Version:

Last Revision Date:

Date Printed

Synopsis:

Author:

Filename:

Draft

1.4

1/22/2010

2/8/2010

Functional design specification for the Ontology cell

Ontology_Design_Oracle

Workplace Design Specifications

Revision History

Author	Date	Number	Description
Janice Donahoe		1.0	Initial Version
Mike Mendis	01/12/10	1.4	Modified the format

Table of Contents

1.	INTRODUCTION	4
2.	RELATIONSHIP OF I2B2 ONTOLOGY TO STAR SCHEMA	4
2.1	DATA STORAGE	4
2.2	ONTOLOGY TABLE	4
2.3	DEFINITION OF FIELDS IN ONTOLOGY TABLE.....	5
2.3.1	<i>c_hlevel</i>	5
2.3.2	<i>c_fullname</i>	5
2.3.3	<i>c_name</i>	5
2.3.4	<i>c_synonym_cd</i>	5
2.3.5	<i>c_visualattributes</i>	5
2.3.6	<i>c_totalnum</i>	5
2.3.7	<i>c_basecode</i>	5
2.3.8	<i>c_metadataxml</i>	5
2.3.9	<i>c_facttablecolumn</i>	5
2.3.10	<i>c_tablename</i>	5
2.3.11	<i>c_columnname</i>	5
2.3.12	<i>c_columndatatype</i>	5
2.3.13	<i>c_operator</i>	5
2.3.14	<i>c_dimcode</i>	5
2.3.15	<i>c_comment</i>	5
2.3.16	<i>c_tooltip</i>	5
2.3.17	<i>update_date</i>	5
2.3.18	<i>download_date</i>	5
2.3.19	<i>import_date</i>	5
2.3.20	<i>sourcesystem_cd</i>	5
2.4	DEFINITION OF TERMS.....	5
3.	SAMPLE ONTOLOGY QUERIES	5
3.1	QUERY SAMPLE FOR DIAGNOSES	5
3.2	QUERY SAMPLE FOR PROBLEMS	9
3.3	QUERY SAMPLE FOR LABS	9

1. INTRODUCTION

This document describes the requirements, technical functionality and the intended capabilities of the project management routines in the i2b2 Web Client. It is to be used as a guideline and continuing reference point as the developers write the code and quality assurance writes the test plans.

2. PROJECT MANAGEMENT CELL CONCEPTS

2.1 OBJECTIVES OF PROJECT MANAGEMENT CELL

Information in the project management cell is related to the setup and maintenance of the hive, projects, users and security.

2.2 IDENTIFICATION OF USERS

Clinical Researcher

- Member of the research team who is setup with access to the project in i2b2.
- Their access role is “**USER**”.
- Can edit their user profile.

Manager of Clinical Researcher

- Manager of the research team.
- Their access role is “**MANAGER**”.
- Can create and edit users associated to their project.
- Can create and edit project related information.

Administrator

- May or may not be part of the research team.
- Responsible for the administrative tasks related to the i2b2 Workbench.
- Their access role is “**ADMIN**”.
- Can create and edit users associated to any project.
- Can create and edit all projects.
- Can create and edit hive information.

2.3 DEFINITION OF TERMS

TERM	DEFINITION
Cell	
Environment	
Global	
Group	
Hive	
Layout	
Pagelet	
Profile	
Project	
Roles	
Users	
Cell	
Environment	
Global	
Group	
Hive	
Layout	
Pagelet	
Profile	
Project	
Roles	

3. REQUIREMENTS

3.1 FUNCTIONAL REQUIREMENTS

To assist with workflow and overall ease of use, individuals will now be able to easily save data and parameters for the hive, projects and users. An outline of

what can be stored in the PM database and the related functionality is listed below.

Global Setup

In the i2b2 Project Management, global data refers to information that is not specific to any one hive or project. The information setup here is used by all.

Global Parameters

Parameters that are to be used by all hives and projects will be defined in the global parameters pagelet. The data will be stored in the **GLOBAL_PARAMS** table.

Users Setup

In order to use the i2b2 application a user has to be setup in project management and given access to one or many projects. In addition, variables can be setup for a user that are to be used across multiple projects or specific to one project.

User Data

General information about the user will be stored in the **PM_USER_DATA** table. The user_id will be referenced in other tables such as **PM_USER_PARAMS** and **PM_PROJECT_USER_ROLES**.

Data Type Code

The codes are used by the parameters table to determine what the object is.

DATATYPE_CD	DATATYPE_DESC
T	Text that is less than 2000 characters
M	Reference to Text file (>2000Characters)
C	Reference to binary file
N	Numeric (Float)
D	Date (yyyy-MM-ddTHH:mm:ss)
I	Integer
B	Boolean (T/F)
IP	(Reserved)
EP	(Reserved)

RTF	Reference to RTF File
XLS	Reference to Microsoft Excel
XML	Reference to XML file or blob
DOC	Reference to Microsoft Word Document

User Parameters

Parameters entered here are not specific to any one project. The information will be stored in the **PM_USER_PARAMS** table.

User Project Roles

The Roles pagelet will capture the information and it will be stored in the **PM_PROJECT_USER_ROLES** table. Each user will have at least two roles per user_ID and product_ID combination. These two roles can be further defined as a Data Protection role and a Hive Management role.

The data protection role/path establishes the detail of data the user can see while the hive management role/path defines their level of functionality the user has in a project.

Data Protection Track	
DATA_PROT	Can see identified data. (formerly "protected_access")
DATA_DEID	DeID text. Can see blobs <i>Note:</i> if there is any identified data it will be encrypted.
DATA_LDS	Can see HIPAA defined "Limited Data Sets" Can see PDO objects without the blob
DATA_AGG	Can see aggregate data
DATA_OBFSC	Can see obfuscated aggregate data

Hive Management Track	
ADMIN	Can see anything in the Hive except identified data

MANAGER	Manage users in project that they are a part of. Change project parameter settings
USER	Regular member Change their user settings

Note: additional roles can be added to the **PM_PROJECT_USER_ROLES** table but there will not be any recognized hierarchy to those roles.

Hive Setup

Users have the ability to setup multiple hives.

Hive Data

The hive data pagelet captures information about the domain ID and domain name as well as the environment and help URL. The information will be stored in the **HIVE_DATA** table.

Hive Parameters

The information will be stored in the **HIVE_PARAMS** table.

Cell Data

The cell data pagelet captures information and registers the cells associated to the hive. The information will be stored in the **CELL_DATA** table.

Cell Variables

The cell variables pagelet captures parameters for each cell associated to the hive. The information will be stored in the **CELL_PARAMS** table.

Project Setup

Multiple projects can be setup.

Project Data

The project data pagelet captures general information about the project. The information will be stored in the **PROJECT_DATA** table.

Project Parameters

The project parameters pagelet captures general information about the project. The information will be stored in the **PROJECT_PARAMS** table.

Project Users

The project users pagelet captures the users who are allowed access to the project. In addition, the user's role and ultimately level of access on the project will also be defined. The information will be stored in the **PROJECT_USER_ROLES** table.

Project User Parameters

User parameters that are specific to the project will be entered via the project user parameters pagelet. The information will be stored in the **PROJECT_USER_PARAMS** table.

3.2 DESIGN REQUIREMENTS

Individual pagelets will be used to capture and store information regarding the hive, project and users. These pagelets will be grouped onto a page based on the workflow and function. For instance, the project setup page will contain the general project information, project parameters, user roles and user parameters specific to the project.

Navigation

A navigation bar will remain on the left side of all the pages. This can be used if a user wants to quickly access a particular pagelet.

Pagelets

Individual pagelets will be used to capture information throughout the project management module. These pagelets will be specific to the task and can be expanded, collapsed and moved around the page.

Home Page

The home page will be a snapshot of all the key areas of PM. Particularly, the hive, project and user management.

Profile Page

The profile page will be a snapshot of particular function of PM. Each major area of the PM will contain a profile page. Each of the pagelets can be expanded or collapsed.

Setup Page

The setup page will be grouped by a particular function of PM. Each of the pagelets can be expanded or collapsed.

4. TABLES

4.1 GLOBAL TABLES

PM_GLOBAL_PARAMS

The PM_GLOBAL_PARAMS table contains parameters that are defined for the entire PM Cell. They are not specific to a hive or project. There will be one table per PM.

- The PM_GLOBAL_PARAMS table is commonly used in a production system to specify default values for various users.

PM_GLOBAL_PARAMS		
PK	ID	int

	User_id	Varchar(50)
	Param_name_cd	Varchar(50)
	Project_path	Varchar(50)
	Value	Varchar(255)
	Datatype_cd	Varchar(50)
	Can_override	Int

PM_USER_DATA

- Information about the user
- The **PM_USER_DATA** table has one row per user and data with special tags that can occur one time per user is in that row.
- **user_ID** is associated with a humanly legible name in the format that they user prefers.
- **password** is the MD5 hash for the users password.
- **email** is the users preferred email address.

PM_USER_DATA		
PK	user_ID	varchar(50)
	full_name	varchar(255)
	password	varchar(255)
	email	varchar(255)

A “_params” table contains name-value pairs associated with that user that can occur more than once or vary from user to user. An “@” in place of a **user_ID** means that the parameter applies to all users not explicitly listed.

PM_USER_PARAMS

- User parameters.
- **ID** is a auto generated number
- **user_ID** is associated with a humanly legible name in the format that they user prefers.
- **Param_name_cd** is associated with the object in the code_lookup table

PM_USER_PARAMS

PK	ID	int
	User_id	Varchar(50)
	Param_name_cd	Varchar(50)
	Value	Varchar(2000)
	Datatype_cd	Varcar(50)

4.2 HIVE TABLES

PM_HIVE_DATA

- Information about the hive.
- Can have multiple hives entered in one PM but only one can be **active** at any one time.
- The name in the **domain_name** column represents the hive domain identifier.
- **domain_ID** column is expected to be highly unique (at least 20 character random sequence) to make it unique across enterprises.
- Each **domain_ID** row contains the **domain_name** for the hive and the environment_CD.
- The **environment_CD** can be easily switched by changing the active configuration.

PM_HIVE_DATA		
PK	domain_ID	varchar(50)
	environment_CD domain_name helpurl active	varchar(50) varchar(255) varchar(255) bit

The enumerated values that represent the environment are:

PRODUCTION

TEST

DEVELOPMENT

STOPPED

Temporarily down

INACTIVE

Undefined down period

ARCHIVED

Not active, not anticipated to be restarted in the future.

PM_HIVE_PARAMS

- The **PM_HIVE_PARAMS** table is one of several “_params” tables in the PM cell.
- The “_params” tables are a manner to express locally defined *name-value* pairs for various entities.
- These parameters are associated with the various “**domain_ID**’s” from the **PM_HIVE_DATA** table.
- This table allows users to specify *name-value* pairs associated with various **PM_HIVE_DATA** configurations.
- These parameters are not specific to any project or user.

PM_HIVE_PARAMS		
PK	ID	Int
	Dmain_id	Varchar(50)
	Project_path	Varchar(255)
	Param_name_cd	Varchar(50)
	Value	Varchar(2000)
	Datatype_cd	Varchar(

PM_CELL_DATA

- Information about a cell.
- The **PM_CELL_DATA** table has one row for each **cell_ID**.
- A hive may have several cells of the same type, but they will be distributed to the projects according to their project path.
- In version 1.x of the Hive software, the XML will only return the cell which is the most specific for that project.

EXAMPLE:

If 3 ONT cells exist:

1. one with project “/hive”
2. one with project “/hive/asthma”
3. one with project “/hive/asthma/snm0”

Only the ONT cell specified by “/hive/asthma/snm0” will be returned for project “snm0”. It essentially allows certain projects to be “diverted” from the mainstream cells.

PM_CELL_DATA		
PK PK	cell_ID project_path	varchar(50) varchar(255)
	name URL method_CD can_override	varchar(255) varchar(255) varchar(50) bit

PM_CELL_PARAMS

- Cell specific parameters.
- The **PM_CELL_PARAMS** table follows the same rules for the return of specific cell-associated *name-value* pairs.
- Commonly used in a production system to specify default values for various users.

PM_CELL_PARAMS		
PK	ID	varchar(50)
	Cell_id	Varchar(50)
	Project_path	Varchar(255)
	Param_name_cd	Varchar(50)
	Value	Varchar(2000)
	Datatype_cd	Varchar(50)
	can_override	bit

4.3 PROJECT TABLES

PM_PROJECT_DATA

- Information about the project
- The **PM_PROJECT_DATA** table has one row per **project_id** and data with special tags that can occur one time per project is in that row.
- The **project_ID** is a unique identifier.
- **project_name** is a short human legible name for the project.
- **project_key** is the MD5 hash for a project key used to encrypt data.
- **project_wiki** contains a URL for the project wiki.
- **project_path** is used in queries to find the value for cell and global parameters. Below is a more detailed explanation and example.
- **project_description** is a long description that can be searched.

PM_PROJECT_DATA		
PK	project_ID	varchar(50)
	project_name	varchar(255)
	project_key	varchar(255)
	project_wiki	varchar(255)
	project_path	varchar(255)
	project_description	varchar(255)

project_path

- Represents a special string in the format *"/Project_name"/sub-project_name"/* etc.
- Used primarily to manage default parameters that the users will see in regards to the interactions they have with a cell.
- Will be used in queries to find the value for cell and global parameters.
- The hive is initialized with default values for all parameters with a blank/null project, following which the params may be overridden by adding a project.
- When the query returns, only one value for each param is returned, but that is the most specific available.

EXAMPLE:

project_path	Value
/	Overall hive default
/ASTH	Asthma default
/HTN	Hypertension default

If the above table was queried by a member of the asthma project who is not a member of the SNM0 sub project then “Asthma default” would be obtained. If the project “MDD” was to query the table as a member of the major depression project, the value of “Overall hive default” will be obtained because a more specific entry does not exist for the user.

PM_PROJECT_PARAMS

- Project specific parameters
- Each project can have its own set of parameters

PM_PROJECT_PARAMS		
PK	ID	Int
	Project_id	Varchar(50)
	Param_name_cd	Varchar(50)
	Value	Varchar(2000)
	Datatype_cd	Varchar(50)

PM_PROJECT_USER_ROLES

- Users associated to the project and their respective role.
- Data is specific to the project.
- The PM_PROJECT_USER_ROLES table will have at least two roles per user_ID and product_ID combination.
- “Data Protection Track” role establishes the detail of data that may be seen by the user.
- “Hive Management Track” role establishes the amount of control a user has in a project.
- The roles are in hierarchical order, similar to the Data Protection Track, so that the roles on top gain the permissions from the ones below.
- Although the table will only contain the role for the highest level of detail the user can see, other roles to see less detailed data will also be automatically granted.
- If a cell requires other unique roles for a user, these can be added to the **PM_PROJECT_USER_ROLES** table, but there will be no recognized hierarchy to those roles.

- For roles that span across all projects, the **project_ID** column in the **PM_PROJECT_USER_ROLES** table gets an "@" sign.
- For roles that span across all users (for a project), the **user_ID** column in the **PM_PROJECT_USER_ROLES** table gets an "@" sign.
- At a minimum, a user/project combination will have a role from at least each of the two tracks listed below:

Data Protection Track	Hive Management Track
DATA_PROT	ADMIN
DATA_DEID	MANAGER
DATA_LDS	USER
DATA_AGG	
DATA_OBFSC	

PM_PROJECT_USER_ROLES		
PK	project_ID	varchar(50)
PK	user_ID	varchar(50)
PK	user_role_CD	varchar(50)

PM_PROJECT_USER_PARAMS

- User parameters that are specific to the project

PM_PROJECT_USER_PARAMS		
PK	ID	int
	Project_id	Varchar(50)
	User_id	Varchar(50)
	Param_name_cd	Varchar(50)
	Value	Varchar(2000)
	Datatype_cd	Varchar(50)

4.4 ACCESS/RESTRICTION TABLES

PM_ROLE_REQUIREMENT

- In order to assign the permissions to edit the PM tables, entries are made into the PM_ROLE_REQUIREMENT table as shows below.
- There is a column to record read and a column to record write permissions.
- The Manager permission is only allowed in tables that have a project_id or a project_path.
- Each table name and column name are specified in the table.
- If there is an @ in the table_CD or the column_CD column for a specific code, that is similar to a wild card character and implies that the code can be used in any column with a similar name in any table, or any column in the table.
- name_char can be used for a description of the purpose and function of a column (varchar datatype for lookup/searching capabilities).

PM_ROLE_REQUIREMENT		
PK	table_CD	varchar(50)
PK	column_CD	varchar(50)
PK	read_hivemgmt_CD	varchar(50)
PK	write_hivemgmt_CD	varchar(50)
	name_char	varchar(2000)

4.5 AUDIT TABLES

Each table in the project management database has a limited auditing of its own management through the change management columns. These columns record the date-time of the initial entry, the date-time of the latest change, the type of change (delete, update, create), and the user_id of the person who performed the change. The columns are filled in when changes are made by the web services. There is no guarantee that the columns are filled in during direct database access.

Transaction columns		

	entry_date change_date changeBy_char status_CD	datetime datetime varchar(50) varchar(50)
--	---	--

The codes in the status_cd column are:

D = delete

U = update

C = create

Note:

These columns can be made into a complete audit trail by adding “read” to the changes recorded, and by creating a new record every time a CRUD transaction is performed. The primary key should be changed to include all of these columns, and only the latest change should be read.

4.6 MISCELLANEOUS TABLES

PM_CODE_LOOKUP

- Any of the “_CD” columns have descriptions available in the **PM_CODE_LOOKUP** table.
- This table is available for client applications to obtain the list of codes that may be entered by the used for the column of a specific table.
- name_char has the descriptive name of the code.
- **lookup_keys_char** is a string with a bar-type-delimiter that allows strings to be used to lookup subsets of codes.
- If there is an “@” in the **table_cd** or the **column_cd** column for a specific code, that is similar to a wild card character and implies that the code can be used in any column with a similar name in any table, or any column in the table.

PM_CODE_LOOKUP		
PK	table_CD	varchar(50)

PK PK	column_CD code_CD	varchar(50) varchar(50)
	name_char lookup_keys_char	varchar(2000) varchar(2000)

PM_PERMISSIONS

- Recording Permissions and Restrictions will occur through the “Permissions” table.
- **ID** is a unique code for the Permission, and could be an IRB number (for example).
- The **name_Char** column specifies what attribute of the Permission is represented in the Permission value column.
- A permission in this table may be tied to a project through the **PM_PROJECT_PARAMS** table, indicating that this permission was applied to create or restrict the project, may be tied to a user through the **PM_USER_PARAMS** table, indicating that this user has permissions (for example as tied to a compliance course that was taken by the user), or may be tied to the User/Project combination through the **PM_PROJECT_USER_PARAMS** table, indicating the user has that permission on a specific project, such as the ability to create a mini-database from a larger database.
- The Object_ID is a linkage to the metadata objects.

PM_PERMISSIONS		
PK	permission_ID	varchar(50)
	permission_name permission_description	varchar(255) varchar(2000)
	permission_activation_date	datetime
	permission_expiration_date	datetime
	object_CD	varchar(50)

PM_PERMISSIONS_PARAMS		
PK	ID	varchar(50)
	permission_ID	varchar(50)
	param_name_CD	varchar(50)
	value	varchar(2000)
	activation_date	datetime
	expiration_date	datetime
	datatype_CD	varchar(50)

	object_CD	varchar(50)
--	-----------	-------------