i2b2 Cell Messaging

Project Management (PM) Cell

Table of Contents

2. Document Version History

Date	**Version**	**Description**	**Author(s)**
August 2, 2007	1.0	First Version	Vivian
January 3, 2008	1.1	Additional explanation	Shawn

3. Introduction

This document gives an overview of i2b2 cell messaging as well as a more detailed description of message formats specific to the Project Management (PM) Cell.

3.1 The i2b2 Hive

Informatics for Integrating Biology and the Bedside (i2b2) is one of the sponsored initiatives of the NIH Roadmap National Centers for Biomedical Computing (http://www.bisti.nih.gov/ncbc/). One goal of i2b2 is to produce a comprehensive set of software tools to enable clinical investigators to collect and manage their project-related research data, including clinical and genomic data; that is, a software suite for the modern clinical research chart. Since different applications from different sources must be able to communicate with each other, a distributed computing model is needed, one that integrates multiple web-based applications in a standardized way.

The i2b2 hive and associated web services are the infrastructure used to create this integration. The hive is comprised of a collection of cells representing unique functional units. Cells in the hive have an array of roles, such as data storage, data analysis, ontology or identity management, natural language processing, and data conversion, derivation or de-identification. Each cell is a self-contained modular application that communicates with other cells via XML web services. A common i2b2 messaging protocol has been defined to enable the cells to interact with each other, sharing business logic, processes and data.

3.2 i2b2 Messaging Overview

All cells in the i2b2 hive communicate using standard, pre-defined i2b2 XML request and response messages.

A request message is sent from a client to a service and contains information, inside the top-level <request> tag, that allows the service to satisfy the request. The <request> tag contains a <message_header>, <request_header> and <message_body> as shown, below, in Figure 1.

The service sends back a response message, inside a top-level <response> tag, which informs the client about the status of the request and may also contain the actual results. The <response> tag contains its own <message_header>, <response_header> and <message_body> and it may optionally echo the request's <request_header> as shown, below, in Figure 1.

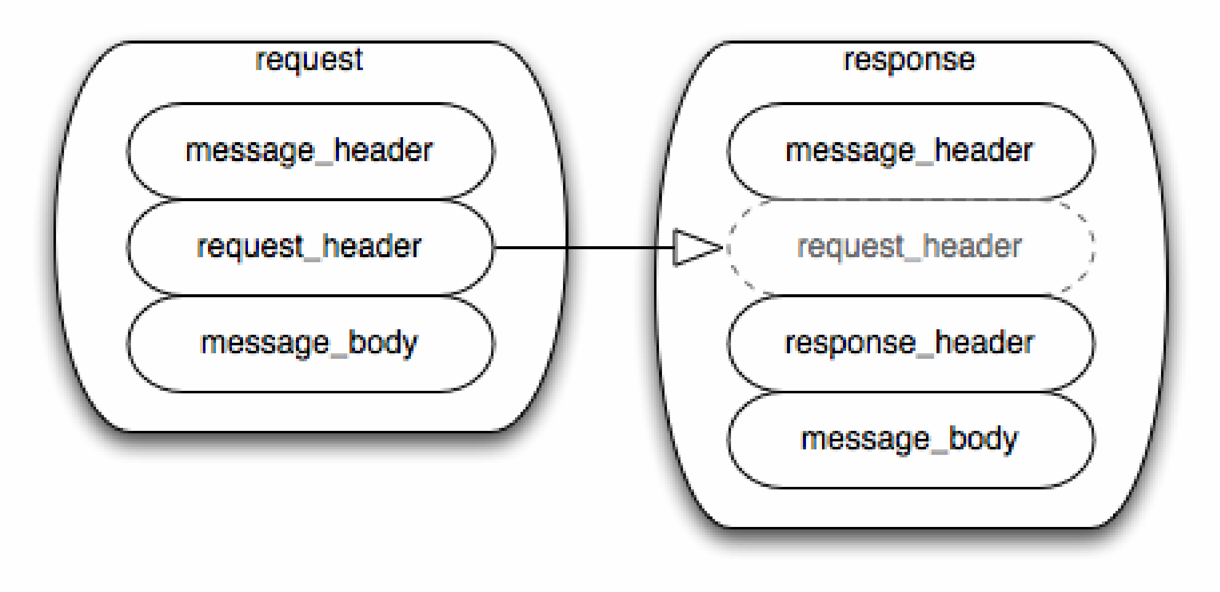


Figure 1: The basic structure of a request and response message. The request_header in the request can be echoed in the response.

3.2.1 Message Header

All requests are sent using a <request> tag and responses are returned using a <response> tag. The same <message_header> tag is used for both. Both request and response messages contain this <message_header> tag which has control information such as sending application, receiving application and message type.

3.2.2 Request Header

The request must contain a <request_header> tag which includes information about how to process a request such as the amount of time it is willing to wait for a response. The <request_header> tag may optionally be echoed back in the response.

3.2.3 Response Header

The response must include a <response_header> tag which includes general information about the response such as status and error messages or where to look for the results if they are not included with the response.

3.2.4 Message Body

Both request and response messages contain a <message_body> tag which may contain any well-formed xml. Individual cells may define cell-specific XML that will be put inside <message_body> tag. This cell-specific XML need not extend the i2b2 message schema since the i2b2 schema will allow insertion of tags from any namespace into the <message_body> tag.

3.3 i2b2 XML Schema Definitions

The i2b2 XML schema consists of three XSD files:

- **i2b2.xsd**
  This schema defines the type for the <message_header> and <message_body> tags described, in sections 0 and 0. This schema is included in the i2b2_request.xsd and the i2b2_response.xsd.

- **i2b2_request.xsd**
  This schema defines the type for the top-level <request> tag and the <request_header> tag described in section 0. It is used for validating i2b2 request messages.

- **i2b2_response.xsd**
  This schema defines the type for the top-level <response> tag and the <response_header> tag described in section 0. It is used for validating i2b2 response messages.

More details about the <request>, <response>, <message_header>, <request_header> and <response_header> tags can be found in a separate document describing the generic i2b2 message. The remainder of this document describes the contents of the <message_body> for the Project Management (PM) Cell.

4. The Project Management (PM) Cell Messaging Detail

The Project Management (PM) cell has two basic functions: to control user access to various services and to keep track of where these services are located. In addition, it stores various (possibly personalized) parameters associated with the services. User access is determined by a user's 'role', which is a variable that defines the actions that user may perform within the HIVE. The role indicates how much data to return and whether or not there is access to a particular service. The default role is 'user' and there are other roles, such as 'manager', 'administrator', and 'protected_access', each with its own set of privileges regarding what data can be viewed. An individual i2b2 user may have one or more roles. The HIVE itself is defined by the PM service. The name of the HIVE is defined in the 'domain' variable. A client which has the root location of the PM services as well as the method (REST/SOAP) for the access to the services can obtain the structure of the HIVE through these services.

When a person logs in to the i2b2 workbench, a login screen comes up that requires the username, password and target location to be entered. (The target location is also called the domain name or simply the domain.) A request message is then sent to the PM cell. Each cell contains a mapping of domain names to urls which provide the address for the service in the PM that will authenticate the user. If the domain is not recognized by a cell (does not exist in the predefined lookup table), or the user does not have access to that domain, then the person is not authenticated to use that specific cell. If the domain exists and the user has access, then authentication occurs. In effect, the mapping of domain names to urls provides the security to the authentication process when using each cell of the HIVE, because the domain name must be "registered" with each cell with the address of a HIVE's PM cell, and then referenced before proceeding. The PM cell returns the user's roles for a project to determine their permissions and privileges in each HIVE cell.

The message used by the PM cell for authentication and authorization is **get_user_configuration**.

4.1 Use Case

The PM cell may be accessed either by an i2b2 client or by another i2b2 cell. The actor is either an end user logging into an i2b2 client, or an i2b2 cell needing to verify particular access. The get_user_configuration request message is sent to the PM cell and a response message is returned.

When an end user logs in, the username, password and domain name are entered and are contained in the message header part of the request message sent to the PM cell. The PM cell uses these values to authenticate the user. The values above may be seen to be replicated in the message_body part of the response message, but are not used by the PM cell for authentication. The replicated values are used to offer the information to other client-sided tools using the object.

The password is substituted by an encrypted token for many interactions. Return messages will then not contain the password.

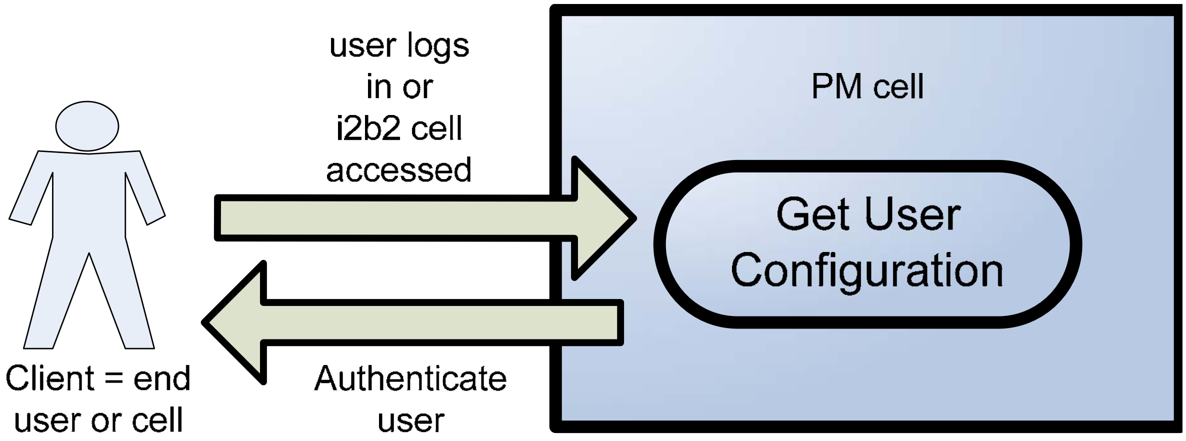
If a user does not have access to the domain or if the domain does not exist, then the request is considered invalid and the invalid user request is returned. If the user does have access to the domain, then the PM cell returns a list of projects the user has access to and the user has the opportunity to pick one in which to continue working. Then a more specific message may be generated to obtain data only relevant to that project.

When another cell accesses the PM server, the purpose is to see what roles exist for this particular user for this cell. The username, password and domain are still in the message header. The body of the get_user_configuration message may include the name of the project being accessed. If the user has a role that allows them to access the cell, a valid response message is returned.

In summary, the PM server performs the following steps via the get_user_configuration message:

a. Authentication: Verifies that the user is valid and is associated with the domain provided. A blank project will return all the domains for the user.

b. Authorization: Gets a list of cells and roles available for this client from the PM cell and return them in the response



.

4.2 Messages

4.2.1 get_user_configuration Request Message

```
<message_body>
        <pm:get_user_configuration>
                 <project>demo</project>
         </get_user_configuration>
 </message_body>
```

A get_user_configuration message is sent by a client application or another cell service to get a list of cells and/or roles associated with that client. The optional attribute ‘project’ can be used to specify for which project the PM cell should return data. If the attribute is blank, then all projects that the user has access to are returned.

```
<request>

            <request>

                        <message_header></message_header>

                        <request_header></request_header>

                        <message_body></message_body>

            </request>

</request>
```

1.1.1 get_user_configuration Response Message

4.2.2.1 Response message for a valid user request

In the PM response message, there are optional ‘param’ XML elements in various locations of the XML. These can be used to describe additional parameters that may exist for different users, projects or global data. For example, if there is a need to prevent a user from getting access to particular data, the user param field could be used to hold a security alert. Or the project parameter could hold the name of a project sponsor or surrogate. The param tags are generic XML tags for adding more information to the messages.

```
<message_body>
     <configure>
         <environment>PRODUCTION</environment>
         <helpURL>http://i2b2.org/help</helpURL>
         <user>
             <full_name>Demo User</full_name>
             <user_name>demo</user_name>
             <password>demouser</password>
             <domain>demo</domain>
             <project id="Demo">
                 <name>Demo Group</name>
                 <key>Demo</key>
                 <wiki>http://i2b2.org/wiki/</wiki>
                 <role>USER</role>
             </project>
         </user>
         <cell_datas>
             <cell_data id="PM">
                 <name>Project Management</name>
                 <url>http://i2b2.org/PMService/</url>
                 <method>SOAP</method>
             </cell_data>
             <cell_data id="ONT">
                 <name>ontology</name>
                 <url>http://i2b2.org/OntologyService/</url>
                 <method>REST</method>
                 <param name="OntMax">200</param>
                 <param name="OntHiddens">false</param>
                 <param name="OntSynonyms">false</param>
             </cell_data>
             <cell_data id="PFT">
                 <name>PFT</name>
                 <url>http://i2b2.org/PFTService/</url>
                 <method>REST</method>
             </cell_data>
             <cell_data id="CRC">
                 <name>previousQuery</name>
                 <url>http://i2b2.org/QueryToolService/</url>
                 <method>SOAP</method>
             </cell_data>
         </cell_datas>
         <global_data>
         </global_data>
     </configure>
</message_body>
```

4.2.2.2 Response Message for an invalid user request

```
<response_header>
        <result_status>
            <status type="ERROR">Supplied password does not match user
password!</status>
        </result_status>
</response_header>
```

5. Glossary

Message Tag & Attribute Definitions For PM_USER.xsd

The Project Management XML schema consists of one XSD file that defines the <message_body> for the entire PM cell, it is called **PM_USER.xsd.**

Header Tag And Attributes Required By PM Cell:

<security> container for authentication

username: a valid user logon

password: the password associated with the user

domain: the server location from which the PM cell will authenticate and obtain roles for the user. When a service is accessed, the cell looks up the domain from a list and associates it with a URL. This URL is used to obtain authentication info. If the domain does not exist in the lookup table, the person is not authenticated.

Request Tag And Attribute:

<get_user_configuration> container for a user data request

project: the project name associated with the user; if null, all projects are returned

Response Tags And Attributes:

<environment> the conditions under which the client is being run, either PRODUCTION, DEVELOPMENT, INACTIVE, TEST, STOPPED or ARCHIVED

<helpurl> points to the URL for the message board displayed by the hive in the above environment.

<user> container for user-specific information

user_name (User_ID): short string user for the user key
full_name: the first and last name of the user

password: the password of the user
project: (defined below)
param: a name|value pair that can be used to give more info about a user
name: a name that describes how the param tag is used

<project>: the projects associated with the user
Id (Project_ID) : a unique string to identify a project
name: human-readable name of the project
key: the key used to manage identified data of he project
wiki: a url for information about the project
role: one tag for each role associated with the project
param: a name|value pair that can be used to give more info about a project
name: a name that describes how the param tag is used

<cell_datas>: the cell_data tags contain information about the cells a user has access to. Note that <cell_datas> cannot be null.
Id (Cell_ID): a unique identifier for the cell
name: the name of the cell
url: the location of the cell
param: a field that can be used to give more info about a cell

<global_data> data which can be used by any of the cells, but is not specific to any individual cell, such as a database connection. Global_data is for additional variables associated with a particular domain.
param: a field that can be used to give more info about a cell