Informatics for Integrating Biology and the Bedside

i2b2

Software Documentation

i2b2 Cell Messaging

File Repository (FR) Cell

Document Version:	*1.7.08-002*
i2b2 Software Version:	*1.7.08*

TABLE OF CONTENTS

DOCUMENT MANAGEMENT

Revision Number	Date	Author	Description of change
1.7.00	03/17/2013	Janice Donahoe	Created 1.7 version of the document
1.7.00-001	08/11/2015	Janice Donahoe	Fixed minor spelling and grammatical errors.
1.7.08-002	10/06/2016	Janice Donahoe	Fixed formatting issues and updated cover page.

1 INTRODUCTION

This document gives an overview of i2b2 cell messaging as well as a more detailed description of message formats specific to the **Data Repository (CRC) Cell**.

1.1 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.bist.nih.gov/ncbc/). One of the goals 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 application 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 of 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.

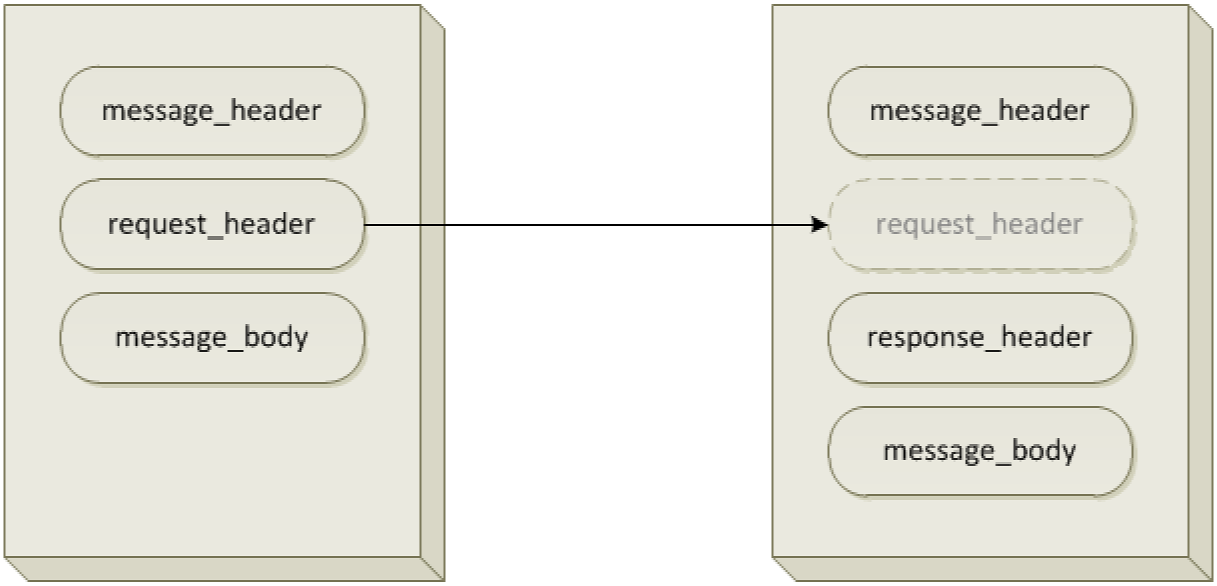
1.2 i2b2 Messaging Overview

All cells in the **i2b2 hive** must communicate using standard *i2b2 XML messages*. This message specifies certain properties that are common to all cells and are essential to the administration tasks associated with sending, receiving and processing 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*>.

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*>.

The following image illustrates the basic top-level elements contained within the request and response messages.



1.2.1 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 the request and response message contain this <*message_header*> tag which has control information such as the sending application, receiving application and the message type.

1.2.2 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.

1.2.3 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.

1.2.4 Message Body

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

1.3 i2b2 XML Schema Definitions

The i2b2 XML schema consists of three XSD files.

1.3.1 i2b2.xsd

This schema defines the type for the <*message_header*> and <*message_body*> tags. This schema is included in the *i2b2_request.xsd* and the *i2b2_response.xsd*.

1.3.2 i2b2_request.xsd

This schema defines the type for the top-level <*request*> tag and the <*request_header*> tag. It is used for validating i2b2 request messages.

1.3.3 i2b2_response.xsd

This schema defines the type for the top-level <*response*> tag and the <*response_header*> tag. It is used for validating i2b2 response messages.

✓ **Additional Resources**

Additional 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 File Repository (FR) Cell.

2 FILE REPOSITORY (FR) CELL MESSAGING DETAIL

The File Repository (FR) cell has two basic functions; (1) to manage files and metadata associated with the file and (2) to send and retrieve the files.

2.1 Use Case

The diagram below depicts common use cases that a user may perform with the FR cell.

2.1.1 Operations

The FR service is designed as a collection of operations, or use cases.

Service	Description

2.2 Messages

2.2.1 recvfile Message

The **recvfile** message is sent by a client application or another cell service to retrieve a file.

ⓘ **Note**

If the file does not need any special role access, then the client can go directly to the URL to retrieve the file; Otherwise the client can send the **recvfile** request message and have the FR server send back the file.

2.2.1.1 recvfile Message Sequence

The **recvfile** message is used by the FR server to retrieve a file. The username, password and project_id from the i2b2 header are used for authentication.

The sequence of events is as follows:

1. The client requests a file (recvfile)

These attributes come from the header and are used by the PM cell for authentication and role access:

Attribute	Description
username	A valid user logon
password	The password associated with the user
project_id	The name of the project location being accessed

2. The FR server performs the following steps:

    a. Authentication: Verify that the user is valid within the domain and the project provided.

    b. Authorization: Based on the role provided from the PM cell, the FR cell will verify the role has permission to access the file.

    c. If the role is allowed access to the file and the file exists, then the file will be returned back to the requester via SOAP response attachment.

2.2.1.2 recvfile Request Message

The “filename” attribute is used to specify the exact location of the file.

Example:

```
<message_body>
    <recvfile_request>
        <filename>/oasis/ABT001B/brain_324.dcm</filename>
    </recvfile_request>
</message_body>
```

2.2.1.3 recvfile Respone Message

The **recvfile** response message contains the status information associated with the file and also the file as an attachment.

Attribute	Description
date	The date the file was created / last modified.
size	The size of the file.
name	The name of the file.
algorithm	The algorithm to verify the file on (MD5, SHA, etc.).
hash	The hash based on the algorithm used.
overwrite	true / false if destination file should be overwritten.

Example:

```
<message_body>
    <recvfile_response>
        <recvfile_response size="3623" name="temp15151.xml" overwrite="true" algorithm="MD5" hash="81100cfcb11417570d613125da90ec65" date="2008-08-20T13:46:36.000-04:00" />
    </recvfile_response>
</message_body>
```

2.2.2 sendfile Request Message

A **sendfile** message is sent by a client application to save a file and the associated data.

2.2.2.1 sendfile Message Sequence

The sequence of events is as follows:

1. The client sends a file directly to the FR cell.

2.2.2.2 sendfile Request Message

The following attributes contain information about the file to be saved and are sent in the sendfile request message.

Attribute	**Description**
date	The date the file was created / last modified.
size	The size of the file.
name	The name of the file.
algorithm	The algorithm to verify the file on (MD5, SHA, etc.).
hash	The hash based on the algorithm used.
overwrite	true / false if destination file should be overwritten.

Example:

```
<message_body>
    <sendfile_request>
        <upload_file size="3623" name="temp15151.xml" overwrite="true" algorithm="MD5"
hash="81100cfcb11417570d613125da90ec65" date="2008-08-20T13:46:36.000-04:00" />
    </sendfile_request>
```

```
</message_body>
```

2.2.2.3 sendfile Respone Message

The FR will return the following response if a valid file request was received:

Example (valid file request):

```
<response_header>
    <result_status>
        <status type="DONE">file was successfully saved</status>
    </result_status>
</response_header>
```

In the cases were an error has occurs it may be due to one of the following possible events:

- The role provided is not valid for the user provided.
- The username / password / project_id returned an error from the PM.
- The file_create_date is not valid.
- The file_modify_date is not valid.
- The file_size was not the same as the actual size.
- Error occurred while saving the file.
- A general server error.
- The file exists and the *overwrite* attribute is set to false.

Example (invalid file request):

```
<response_header>
```

```
        <result_status>
            <status type="ERROR">patient_id provided does not exist!</status>
        </result_status>
    </response_header>
```

3 GLOSSARY

3.1 General Terms

The following table contains terms that are used throughout this document.

UPDATE WITH TERMS RELEVANT TO THIS DOCUMENT. THE FOLLOWING IS AN EXAMPLE OF WHAT CAN BE ENTERED.

Term	Definition
i2b2 Web Client	The i2b2 Web Client will be used when querying your local i2b2 to find the cases (patients) that are to be included in the new data mart.
Project	