

**Terefe, Zelalem**  
**Apr 16, 2020**

<b>Table Content</b>	<b>1</b>
1. Introduction and Overview:	<b>2</b>
2. Architecture and Architectural style:	<b>3</b>
2.1 Components and Connector View	<b>3</b>
2.1.1 Client-Server Style	<b>4</b>
3. Diagrams	<b>5</b>
4. External Interfaces:	<b>6</b>
4.1 Volumes:	<b>6</b>
5. Use Cases:	<b>7</b>
6. Data Dictionary and Naming Conventions	<b>8</b>
7. Test Plan	<b>8</b>
8. Methodology and Cost:	<b>9</b>
8.1 Module View type	<b>9</b>
8.1.1. Decomposition Style	<b>9</b>
9. Performance	<b>12</b>
10. Evolvability	<b>12</b>
11. Feasibility Studies	<b>12</b>
12. Reference	<b>13</b>

## **RingRoad Hospital Clinical Research Chart Cell Solution Architecture**

### **Abstract**

This is a software architecture document for the CRC (Clinical Research Chart) cell. It identifies and explains the important architectural elements. This document will serve the needs of managers , developers and stakeholders to understand the system concepts.

### **1. Introduction and Overview:**

Ring Road hospital has been depending on a CRC cell to save patient's phenotype and genotype information. Even Though the data warehouse has been able to store patients' information it isn't able to handle concepts such as diagnoses, procedures, medications and lab tests. To solve that my team has come up with an architecture solution to make the CRC cell handle diagnoses, producers, medications and lab tests. The CRC will be supported bt a powerful metadata management module(the Ontology Cell). It will be designed with several requirements. The main requirements are:

1. It must be able to hold healthcare information from many different venuses and allow it to be acquired rapidly even if there are hundreds of millions of rows.
2. It must be able to combine with other projects repositories so that it can form large unified repositories
3. It must allow objects that are in genomic data to be stored

All datas of the patients in the CRC are de-identified, except the real patient notes from hospitals. The notes from patients are stored in encrypted form, so that they are only visible to users authenticated with an encryption key. Different technical platforms will be used to build the product, below are list of technologies that will be used:

- Java 2 Standard Edition 5.0
- Hibernate Core 3.1 Object Relational (OR) mapping tool
- Oracle Server 10g database
- Xerces2 XML parser
- JBoss Application server version 4.0.3SP1 and higher
- Spring Web Framework 2.0
- Axis2.1 web service (SOAP/REST)

The primary role of the participants in the CRC system are as follows:

User - Create a query and only allow them access if the user is owner of the query

Project Investigator- Has access to different queries created by different users and can create queries with in the project

The system has a strong security system built within it, users can access the CRC using a user id and password combination which is authorized through the project Management Cell. For this documentation the implementation of the project management cell is out of scope so it will not be explained.

## **2.Architecture and Architectural style:**

Below is the description of the architect as multiple views. Each view brings different attributes of the architecture

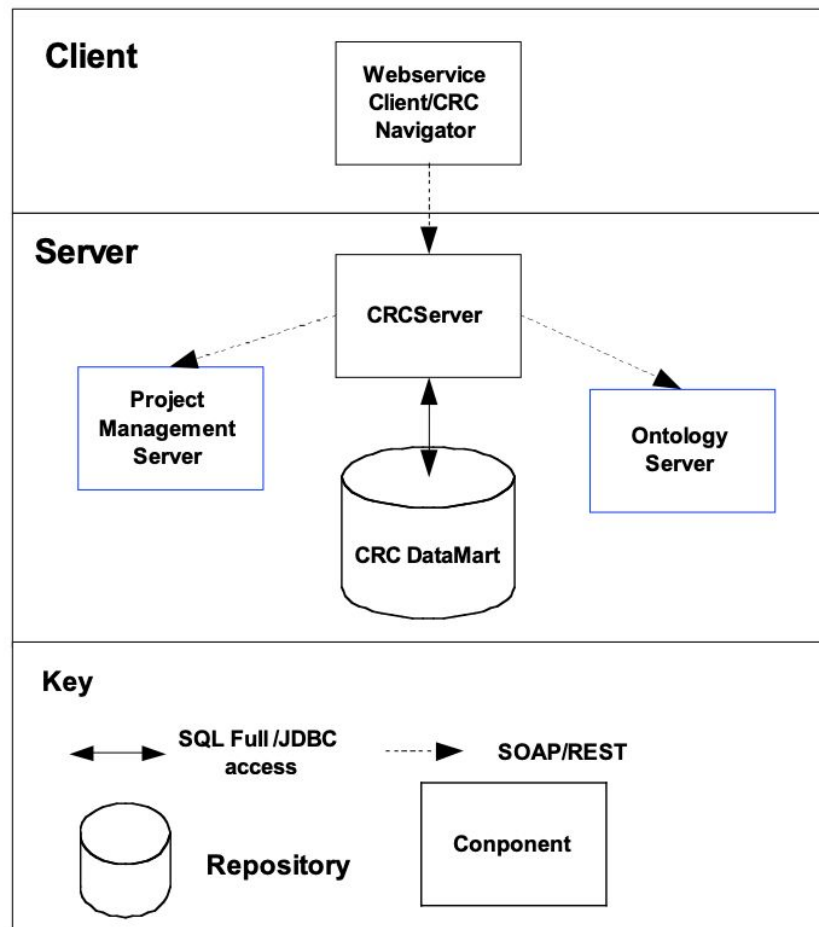
### **2.1 Components and Connector View**

A Component and Connector view represents the runtime instances and also the protocols of connection between the instances. The connectors represent the properties like concurrency,

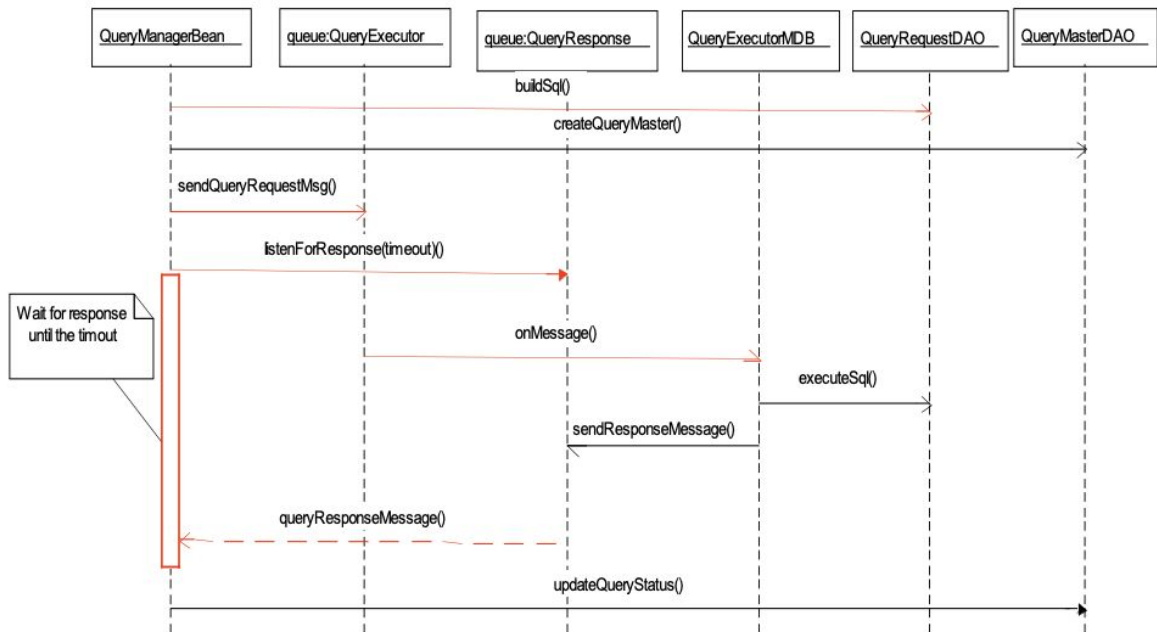
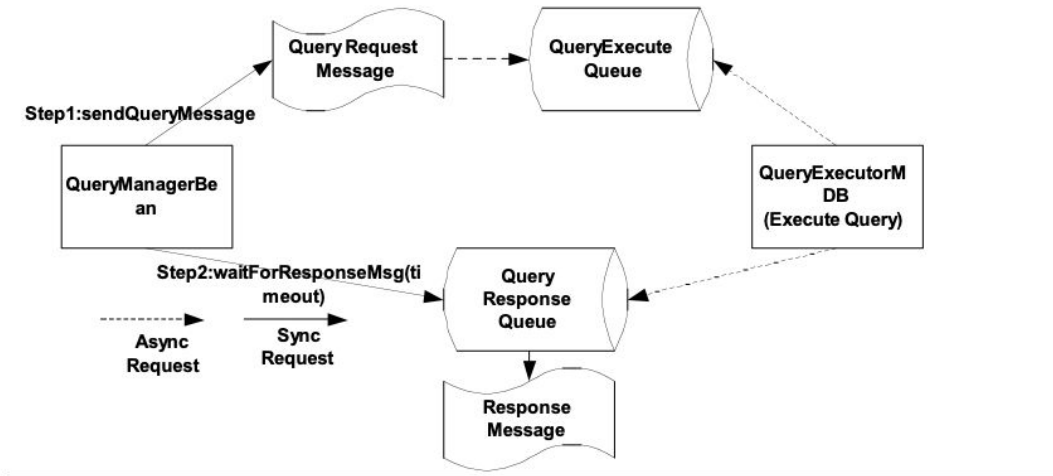
protocols and data flows. Following diagram represents the Component and Connector view for the multi-user installation. As seen below, component instances are shown in additional detail with specific connectors drawn in several notations.

### 2.1.1 Client-Server Style

The CRC system is represented using the C&C Client-Server view.

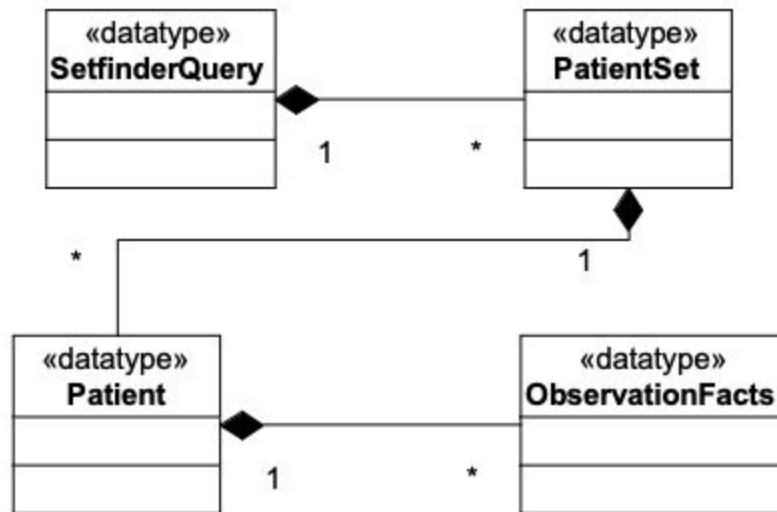


### 3. Diagrams



#### 4. External Interfaces:

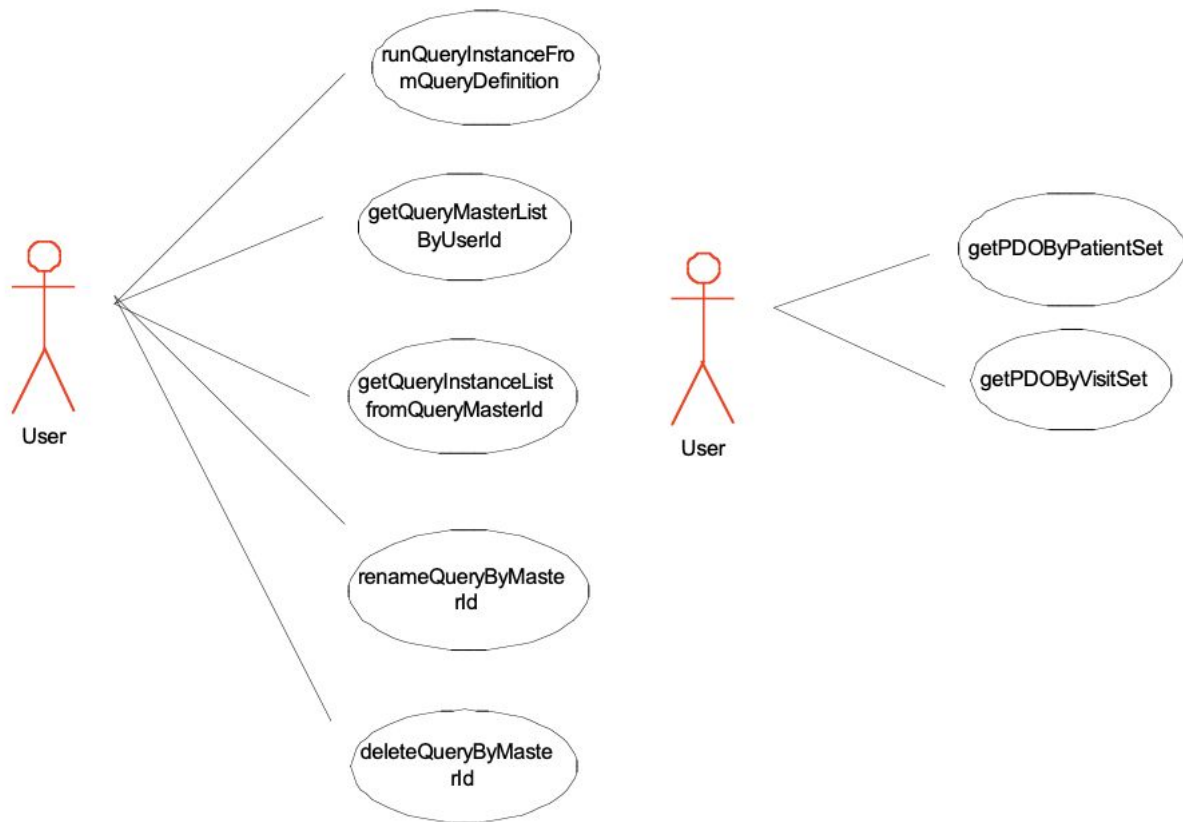
The key data elements related to the CRC system are:



##### 4.1 Volumes:

- Estimated new Set Finder query : 100 a day, with peaks in the morning
- Average PatientSet size 100,000
- CRC registered individual user : about 150

## 5. Use Cases:



- Validate the user by calling the Project Management Cell.
- Save query definition and its generated Sql.
- To scale the appliance and to support long running Sql, the execution of Sql is handled inside the queue.
- If the Sql execution completes before the “result\_waittime\_ms” which is laid out in the request, then the query results are passed within the response message, otherwise the status of the query is passed within the response message.



## **6. Data Dictionary and Naming Conventions**

### **1. Patient Data Object (PDO):**

This Object mirrors the star schema database model of the data mart. It holds patient information such as clinical observations, demographics and provider data.

### **2. Set Finder Query:**

Setfinder queries are used to create a set of patients that satisfy a criteria presented in the query.

### **3. Observation Fact:**

Any observation made on a Patient can be stored as fact information in CRC data mart.

The user can fetch this fact information either via the PDO or Table PDO queries.

## **7. Test Plan**

Writing a command-line script that runs a SQL and checks the output of the following tests:

- Connection to the database engine succeeds
- Database exists
- All the tables exist
- Definition of each table is as expected

It would also be interesting to test insert, updates, deletes and selects encapsulating them in transactions and rolling back at the end, instead of committing

- Check that it's possible to log in
- Check that it fails to log in with wrong credentials
- Check that the dashboard appears with the desired elements in it
- Check for the account menu list
- Check that the system logs out correctly
- Check for all the menu elements
- Check that they can be clicked and the content loads

## **8. Methodology and Cost:**

### **8.1 Module View type**

The module view shows how the system is decomposed into implementation units and how the functionality is allocated to these units. The layers show how modules are encapsulated and structured. The layers represent the “allowed-to-use” relation.

The following sections describe the module view using Decomposition and Uses Style.

#### **8.1.1. Decomposition Style**

The Decomposition view presents the functionality in terms of manageable work pieces. They can be further decomposed to present a higher level of details. The decomposition view identifies modules and breaks them down into sub-modules and so on, till a desired level of granularity is achieved. The “Uses” style shows the relationships between

modules and submodules. This view is very helpful for implementation, integration and testing the system.

## Presentation

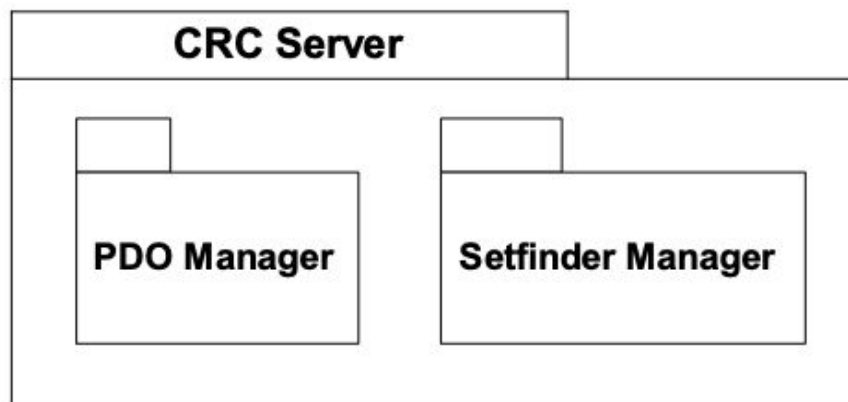
System	Segment
CRC	Setfinder Manager
	PDO Manager

## Element Catalog

### Elements and their properties

Element Name	Type	Description
Setfinder Manager	Subsystem	This subsystem manages user's Setfinder queries. Keep tracks of query information like query definition, its Sql, owner of query, etc. Also the results of query like the patient set, visit set, etc is stored.
PDO Manager	Subsystem	This manages both plain and table Patient Data object queries.

## Diagram



## Use Style

### Presentation

System	Segment
CRC	CRC Module
Setfinder Manager Subsystem	Setfinder Web Service
	Setfinder EJB
	Setfinder DAO
	edu.harvard.i2b2.common
PDO Manager Subsystem	PDO Web Service
	PDO EJB
	PDO DAO
	edu.harvard.i2b2.common

## Element Catalog

### Elements and their properties

Element Name	Type	Description
CRC Module	Module	User Login Module authenticates through PIN Server System with user id and PIN.
Setfinder Webservice	Module	Provides web service interface to Setfinder operations.
Setfinder EJB	Module	Delegates Setfinder requests to DAO layer to perform database operations.
Setfinder DAO	Module	Supports operation like create query master, delete query, saving query definition and its results.
PDO Webservice	Module	Provides web service interface for PDO requests.
PDO EJB	Module	Module to delegate PDO requests to corresponding PDO and to build PDO response

		message.
PDO DAO	Module	Module to query database based on PDO requests.
edu.harvard.i2b2.common	Module	This module provides utility classes to handle JAXB, JNDI, etc.
Persistence Service	Module	Provides SQL interface to database.

## **9.Performance**

The user authentication with the project management cell must be under 10 seconds.

For each query it will join between 3-4 indexes, even representing 100's of millions of Rows to make the query fast

It will use a large analytic database to constantly absorb new data fast

MetaData is used to perform queries to make it faster when dealing with these kind of large analytic database

## **10.Evolvability**

The Clinical Research Chart will handle concepts such as diagnoses, procedures, medications, and lab tests, but the structure of the table gives enough flexibility to expand this to include virtually any kind of observation.

The CRC system is going to be transactional, leveraging the technical platform capabilities. The transaction management model of the J2EE platform will be reused intensively. In the implementation, to support long running set finder queries, transaction management will be manually turned off until the completion of the query.

Application will use both the JDBC calls and Object/Relation mapping tool(Hibernate) to persist data

## **11. Feasibility Studies**

The Software will solely be designed for research purposes only. The plan we proposed to develop this product has a potential financial cost, including adoption and implementation costs, ongoing maintenance costs. A potential productivity decrease may come after adoption of

the system until the physicians and researchers totally become familiar with the software. Changes in workflow can happen because it is a totally new architecture and system development. Software update and hardware replacements are necessary otherwise the software will be vulnerable for an attack by hackers. Due to that there will be a cost to keep the system up to date.

## 12. Reference

[https://www.i2b2.org/software/projects/datarepo/CRC\\_Architecture\\_10.pdf](https://www.i2b2.org/software/projects/datarepo/CRC_Architecture_10.pdf)

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3270933/>