MEDINFO 2007
HISA Ltd Australia
© 2007 The authors. All rights reserved.

Enabling Biomedical Research in Europe: Using the Dutch Experience as a Template

René Verheij[a], Tyrone Grandison[b], Frank Baas[c],

[a]*IBM Global Business Services -Netherlands, Johan Huizingalaan 765, P.O. Box 9999, 1006 CE Amsterdam, The Netherlands*
[b]*IBM Almaden Research Center, 650 Harry Road, San Jose, CA 95120, United States*
[c]*Academic Medical Center, Meibergdreef 9, 1105 AZ Amsterdam, The Netherlands*

Abstract

Medical centers in the Netherlands contain a wealth of patient information that could be leveraged by healthcare researchers to drive breakthroughs, to deliver innovation in healthcare and improve the quality of life for Dutch citizens. However, the sensitivity of this data, the fragmentation of patient records and the complexity of retrieving this information have inhibited advances in this arena. In order to open up clinical data to researchers, the Academic Medical Center (AMC) in Amsterdam has implemented a framework that addresses the problems outlined, mitigates the risk for practitioners and patients and enables medical innovation to proceed. In this paper, we describe the AMC environment, we present the system implemented and highlight the benefits of employing this technological approach. We purport that the AMC model can be used as a template for Biobanks across the world.

Keywords:

Biobank, healthcare research, data integration, data abstraction, privacy

Requirements

The main problems faced by the medical researchers at the AMC were how 1) to acquire data from multiple disparate systems within a reasonable timeframe, 2) to ensure that patient privacy is respected, and 3) to enable natural interaction with the computer systems (i.e. not mandate that medical researchers become computer specialists). These concerns raised three higher-level base requirements for a solution system:

- Data Integration – providing a consistent, global view of a patient, regardless of the locations of the records, the data formats or the database vendor.
- Disclosure Control – ensuring that patient privacy is not violated, organization security policy is adhered to and that proactive compliance is possible
- Usability – providing a simple interface that hides the complexity of the underlying infrastructure and allows researchers to easily query the system.

The deployed solution dealt with all the above problems, by leveraging a subset of IBM's Clinical Genomics Framework (Figure 1).

Query System: Data Discovery and Query Builder

DDQB provides an easy-to-use way of extracting information from data sources by allowing non-IT professionals to create queries using their own terminology. DDQB's central component is the Data Abstraction Model (DAM), which bridges the gap between the technical complexity of databases and the language and concepts that are familiar to end users. DDQB is accessible by means of a standard Internet browser (i.e. Microsoft Internet Explorer, Mozilla Firefox, etc.)

Disclosure Control: Hippocratic Database Technology

HDB receives user queries from DDQB and rewrites the queries so that the database will only return data that is consistent with company policies, applicable legislation, and customer preferences. Data that is not consistent is suppressed or encrypted.

Data Integration: WebSphere Federation Server

WFS uses federation technology to abstract a common data model across diverse and distributed data and content sources, and to access and manipulate them as though they were a single source. The federated system acts as a virtual database in which remote objects can be accessed similar to local tables.

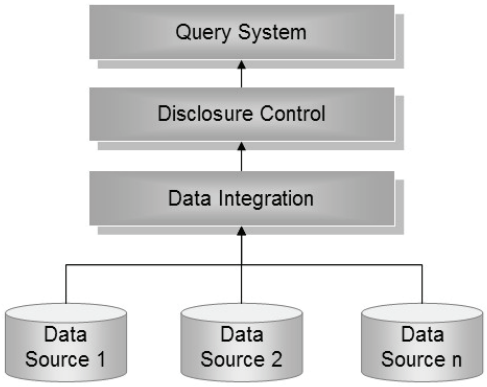

Figure 1 - Solution Architecture

Usage scenario

In a typical scenario, a researcher, let's call him Carlo, wants to retrieve information from systems in the departments of Internal Medicine and Clinical Chemistry in order to perform an analysis on pharmacological treatment of patients with Crohn's Disease. Carlo knows that each of these systems uses a different platform, different terminology and database technology, i.e. Internal Medicine uses Microsoft SQL Server on a Windows platform, and Clinical Chemistry uses a proprietary database on Linux. A further complication is that AMC's corporate policy states that for research purposes, patient numbers should never be shared in an unencrypted form and that medicinal doses should be disclosed on an opt-in basis per patient.

Carlo logs into the DDQB system, and verifies his user id and password with AMC's existing user directory. He pecifies a query through the menu-driven interface of DDQB. Carlo wants to select patients diagnosed with Crohn's Disease, between the ages of 25 and 50 years old, who have been treated with azathioprine. Carlo also instructs the system to retrieve plasma concentrations of liver enzymes, so that he can evaluate the toxicity of this drug for the liver. Fields are organized in a tree structure. Carlo selects the fields that represent the desired information by clicking on a link. Selection values can be chosen from a list, e.g. for the diagnosis field (Figure 2 Selecting patients based on diagnosis2), or entered using the keyboard, e.g. for the age limits. Carlo also select lab tests 'ALAT' and 'ASAT' from the Clinical Chemistry category. The system builds a list of query conditions (Figure 3) and fields to be returned. When Carlo indicates that his specification is complete, the application translates it into an SQL statement and submits this to HDB through its JDBC interface. HDB compares this statement to the policy rules that apply, and determines if Carlo has requested any information that he has not been granted access to. If this is the case, HDB rewrites the SQL statement, eliminating this information from the result list. It also determines at this stage if any information must be encrypted, and modifies the SQL statement accordingly. In this scenario, HDB will rewrite the SQL statement to display encrypted patient numbers and ensure that only patients who have consented will have their medicinal doses disclosed. The resulting statement is then returned to DDQB, which passes it to WFS.

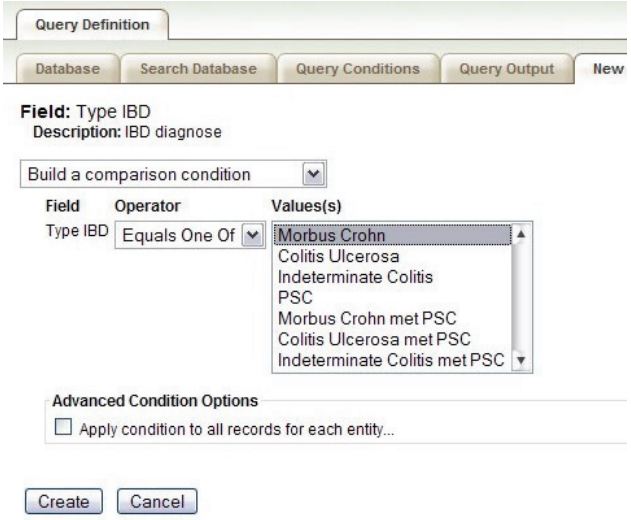

Figure 2 - Selecting patients based on diagnosis

WFS uses its knowledge of the location of the requested information to break up the SQL statement into parts, and submits each part to the appropriate data sources. It combines the result sets returned by the data sources into a single result set. This result set is then returned to DDQB. Finally, DDQB for mats the data and sends it back to Carlo's workstation (5). Note the encrypted patient id values in the 'patientnummer' column. Carlo can inspect the result of his work, store it in DDQB or in a file on his workstation, or feed the retrieved data into an analysis tool. He does not have to perform any manual data manipulation. Specifying and executing a query using the Clinical Genomics application can be completed in minutes. A complex query can be specified and executed in less than an hour, which is a few orders of magnitude faster than previously possible.

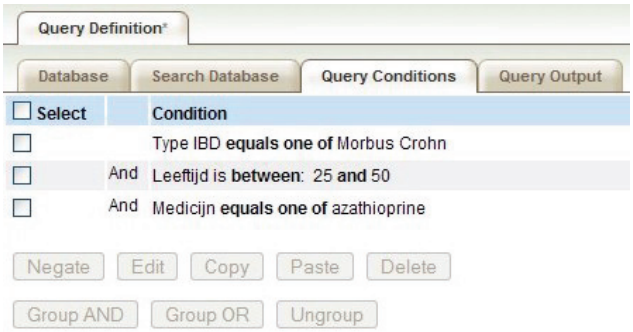

Figure 3 - Query conditions

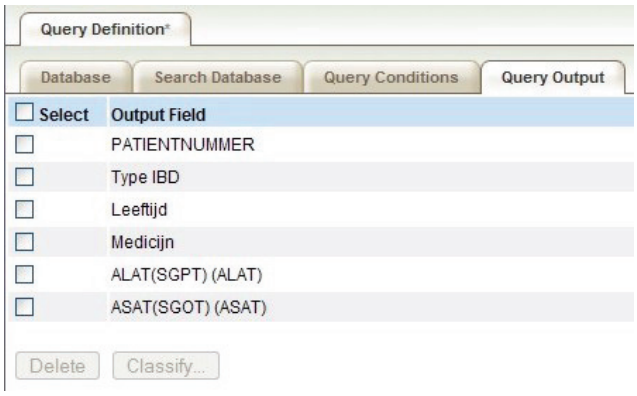

Figure 4 - Query output

System benefits

AMC's new system enables medical researchers to get the data they need efficiently. It is no longer necessary to manually obtain exported data from different systems that must be manipulated manually. This reduces the probability of errors occurring in the data. The researcher is not expected to understand the gory details of all the computer systems used across all the units he interacts with. The dependency on system administrator has been elimi nated. Privacy and security issues, often ignored in the manual process, are enforced (at the technology level) by the system. The process of selecting patient cohorts can now be completed in minutes or hours rather than weeks or months. This is a significant enhancement of researchers' productivity.