Utilizing a Clinical Datamart to Automatically Update a Desktop Database for Leukemia Research

By
Jonathan B. Becker

Capstone Project Paper
May 15, 2003

Capstone Advisor:
Christopher J. Dubay, Ph.D.

School of Medicine

Oregon Health & Science University

Master Of Medical Informatics

Certificate of Approval

This is to certify that the Capstone Project of

Jonathan B. Becker

"Utilizing a Clinical Datamart to Automatically Update a Desktop Database for Leukemia Research"

Has been approved

[redacted]

Professor in charge capstone

5/22/03

Date

Table of Contents

Acknowledgements

I would like to thank Dr. Christopher Dubay, Dr. Rick Press, and Dr. James Brundege for their help with this project. Their efforts made my life much easier. I would also like to thank Dr. Judith Logan for her guidance.

A special thanks goes to my family and friends for their support. Most importantly, I would like to thank Melody for her patience and Jane for smiling at me.

Introduction

Modern healthcare systems generate enormous amounts of data describing the patients they serve. During the middle of the last decade, many of these healthcare organizations began to form repositories to aggregate all data collected by individual department information systems[6]. Traditionally, these data warehouses were utilized primarily for point of care services, public health, financial services and claims administration. Recently, however, many healthcare systems have chosen to use this data for a variety of research needs.

In this paper I will discuss a project that involves the use of Oregon Health and Sciences University's clinical data warehouse known as the Lifetime Clinical Record (LCR). This project addresses the information needs of group of leukemia researchers as they attempt to automatically update an existing desktop database of patient data. I will first give a brief description of some of the terms and concepts involved in the project. I will also discuss some of the current concerns surrounding the use of patient data in healthcare research. Finally, I will describe the methods I have used to address the project's goals and offer suggestions to improve future work.

Data Warehouses and Datamarts

Businesses often utilize a variety of disparate computer systems in order to capture and manage the information needed to operate. Many organizations have integrated this data into a single repository called a *data warehouse*. The benefits of undertaking such a project include better decision-making, employee empowerment, and the ability to

leverage operational data[6]. According to one study, the average return on investment for a data warehouse was 441 percent[6]. While healthcare has traditionally been slow to adopt information practices of the financial and retail industries[13], many healthcare systems are now rapidly implementing clinical data repositories[11] in the face of increased financial competition.

On the surface, a data warehouse is much like a traditional database[10]. Most successful data warehouse projects are implemented using a database management system (DBMS). Most warehouses use relational or object-relational methodology like a traditional database. What distinguishes a data warehouse from a traditional database is how the data is collected and how it is used. A traditional hospital database system such as a laboratory information system (LIS) allows a user to view data from only one patient, possibly many data points collected over long period of time. A data warehouse, however, could be utilized to view whole sets of data from many patients. While an LIS usually contains records with a normalized homogeneous structure, a healthcare data warehouse typically incorporates data from a variety of sources[18]. These sources often have widely variable data models ranging from flat files to images or even whole database systems.

The data warehouse is populated by extracting or capturing data from these original departmental sources. This source data is then transformed and loaded into the warehouse. Transformation of the original data often involves reconciling any differences in semantics or syntax among sources and then mapping each point to its

storage in the warehouse. The following figure represents a model of a typical data warehouse in a healthcare organization:

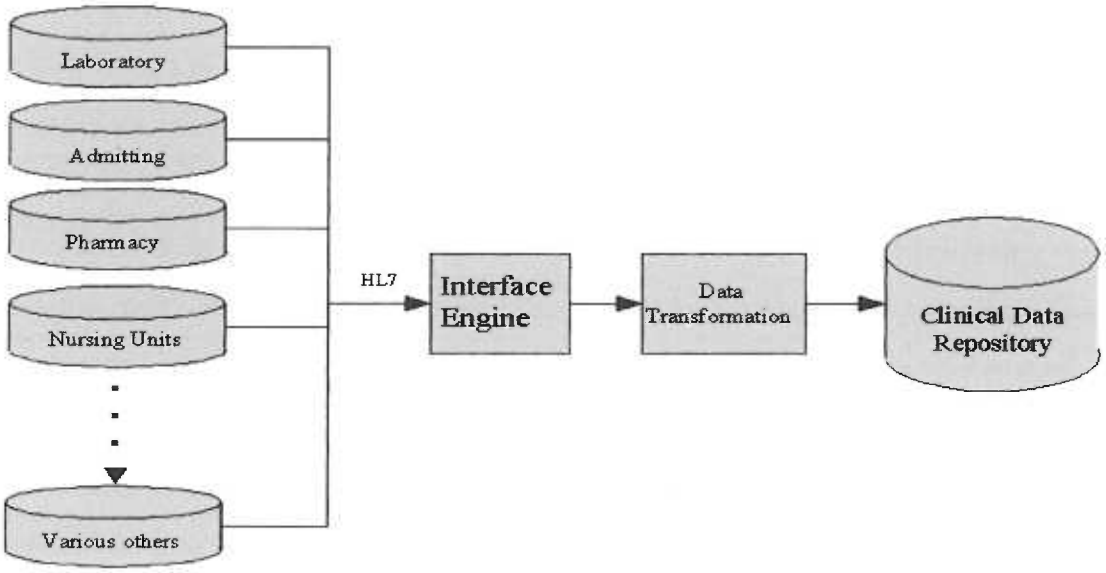


Figure 1 A general model for a healthcare data warehouse

Various individual operational systems export data in HL7 format, which is then transformed and loaded into the data repository. The transformation of data can involve actions such as reconciling differences in syntax or semantics between operational systems, removing redundant data, or mapping to the warehouse data model[18].

OHSU's LCR adheres to the above general model. It is actually implemented on a DB2 database management system. The LCR has a star schema (see appendix 2), its central fact table stores roughly a terabyte of individual data points that may be aggregated in various ways. The fact table is then linked to the dimension tables by one-to-many

relationships. The dimension tables provide the basis for aggregating data from the fact table. A portion of this schema is represented below:

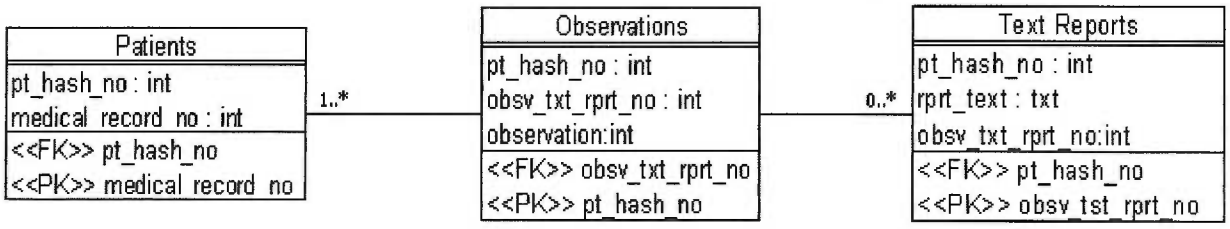


Figure 2 A portion of LCR's schema

While data warehouses are created to meet the information needs of an entire enterprise, a *datamart* is a data set that is built specifically to fulfill the needs of individual groups of users[15]. By definition a datamart is based on a set of user requirements and is designed to solve a specific problem. Individually datamarts are often subsets of data taken from warehouses. Both can be stored in the same system though they are usually physically separate[16].

Leukemia Database

The datamart being created as a result of this project will answer the needs of a group of researchers in OHSU's molecular diagnostics laboratory. The lab is currently studying a group of nearly 400 patients with chronic myeloid leukemia or acute lymphoid leukemia being treated with the drug Gleevec (also known as STI571 or imatinib mesylate). While Gleevec appears to be extremely effective, there have been recent cases of relapse after treatment[20]. It is hoped that by correlating test results with clinical data, factors leading to this resistance might be identified and diagnostic methods may be improved.

This project is centered on an existing database created using Filemaker Pro® database software. This desktop database was originally created to track the results of individual reverse transcription-polymerase chain reaction (RT-PCR) tests in an attempt to correlate them with the patient's hematology data. RT-PCR is a very sensitive method of quanitating levels of a given RNA[2]. A short description of the test methodology is given in appendix 3.

While the RT-PCR results were readily available to the researcher, hematology results were added to the database by periodically importing delimited text files. These text files had to be individually solicited from laboratory information systems analysts. This required substantial time and effort on part of both researcher and analyst. As a result, the database was rarely updated.

To this initial framework of RT-PCR and hematology values, the researchers added tables for three other results of interest: bone marrow morphology, cytogenetics, and BCR-ABL fusion type. The addition of these tests added to the complexity of the problem. Bone marrow morphology and cytogenetics results were stored in a separate operational system for pathology results. Update of the database now required the researcher to contact analysts working in both laboratory and pathology information systems.

The group felt that a well maintained database would be of tremendous value and sought to find a way of automating the update process. Since results for all of the tests were stored in the Lifetime Clinical Record, researchers contacted LCR analysts in attempt to consolidate their efforts. A collaborative project was soon underway to determine how best to provide an automatic update using the LCR. Although there had been previous projects to extract clinical data from the repository for outcomes research, the LCR has been primarily used either as an electronic medical record or for administrative and billing purposes. Many of the tasks the group needed to complete had not previously been prototyped.

HIPAA and Medical Research

The Health Insurance Privacy and Accountability Act (HIPAA) contains broad protections for the privacy of an individual's medical information. HIPAA requires that specific, written authorization be obtained from patients before their medical record can be used for research[1]. This authorization requirement can be altered or waived if the health information meets certain criteria[3]. Most notably, health information can be disclosed without written consent if the request is approved by an institutional review board (IRB) or privacy board[5]. Typically, an IRB would approve such a request only if the release of information involved minimal risk to the individual and the research could not be conducted without the data.

HIPAA also contains a number of provisions that affect how private health information should be transferred. Individual study participants have the right to request an account

of any disclosure of their health information in the last six years. The account must contain the date of each disclosure, who the information was given to, and the purpose of the disclosure. HIPAA also limits researchers to view only personal information that is "necessary" to conduct research. While the exact meaning of this requirement is not explained in great detail, it does force researchers to act with caution[5].

There are separate provisions in HIPAA for patent data that has been stripped of all identifying information. Although requirements have varied slightly since the legislation passed in 1996, the final rules allow academic researchers to create data sets that would not include "facially identifiable"[5] information such as names, addresses, phone numbers, and social security numbers. This would not prohibit inclusion of information such as admission/discharge or service dates, age, medical record numbers, or zip codes as long as the institution provided assurance of privacy.

Although the data sets used to create the updates for the leukemia database do not contain any facially identifiable information, only patients that have signed an unconditional release of information will be included in the datamart update (of the nearly 400 patients being tracked by the molecular diagnostics lab, 145 have signed an unconditional release). This is done, in great part, to protect OHSU from legal liability. A good deal of uncertainty still exists as organizations attempt to institute HIPAA regulations and penalties for violations can be severe. Noncompliance carries both civil and criminal sanctions with penalties of up to $250,000 and imprisonment of up to ten years for each

offense. If data were misused, each patient involved in a data set could seek civil damages of up to $25,000[8].

Constructing the Datamart

The initial task of the project was to determine which data points needed to be included in each datamart. This was accomplished by a thorough review of the structure of the Filemaker database with guidance by the research group. While no schema for this database existed at the project's inception, one was created at this stage in order to assist with building the datamart. This schema is represented in figure 3 on the following page.

Once the necessary data points were identified, each result in the datamart was mapped to its corresponding field in the LCR. This required the warehouse analyst to be provided with a set of unique identifiers, which could be used to locate each test result. These unique identifiers are exported along with each result to the LCR as an attribute-value pair. Appendix 1 has an example of these codes for each result in a complete blood count. Lists of these codes are maintained by the individual departments.

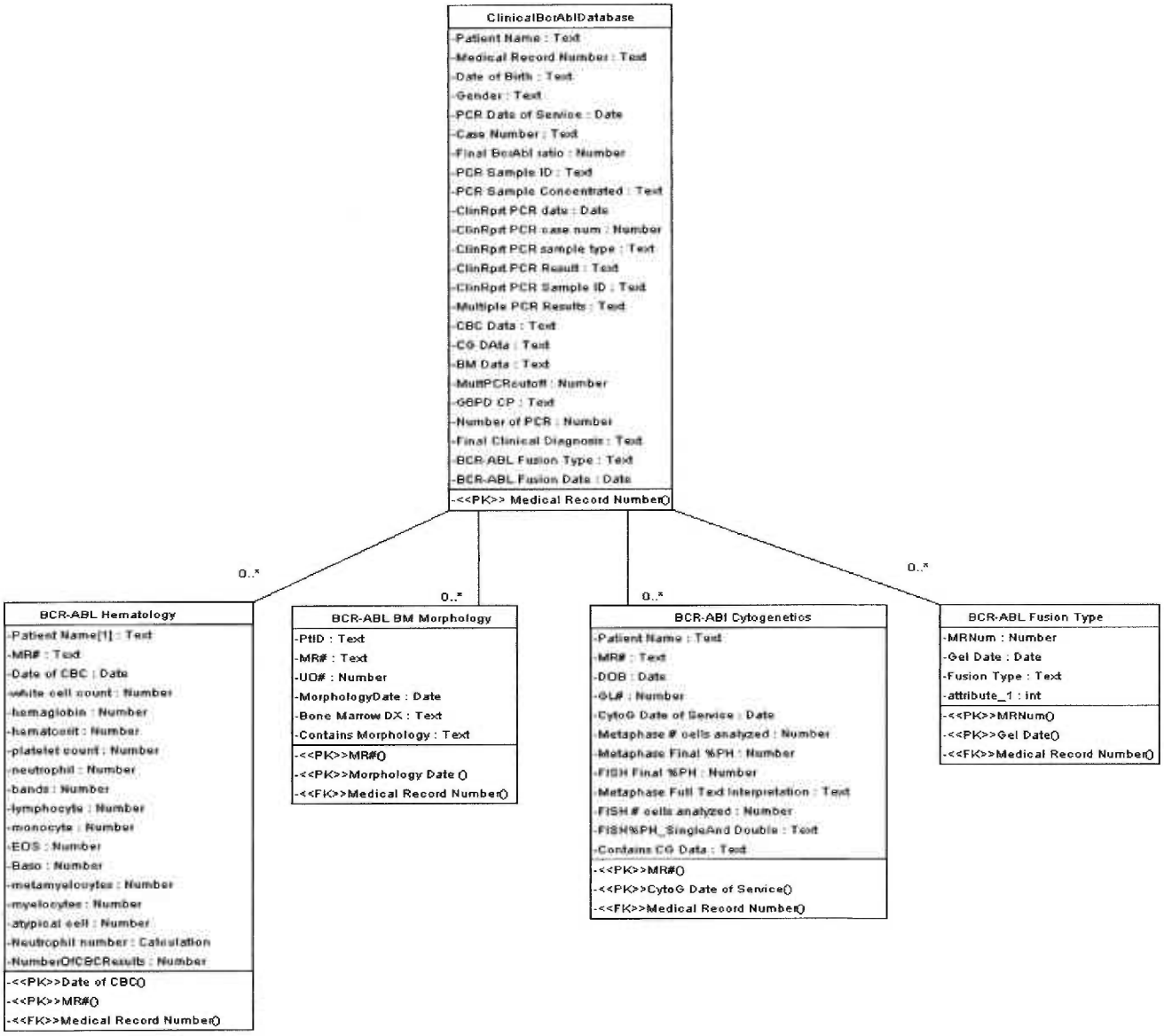


Figure 3 Filemaker database schema

Queries to the LCR were then structured by selecting these data points for patients from the Filemaker database. Two types of reports were generated from these queries: one for results of tests that are stored as numeric data and one containing results of dictated text. The reports were scheduled to be created once a month. They were stored in the form of delimited text files and transferred upon creation to an FTP account for access by the research group. The following figure is a use-case diagram depicting this process:

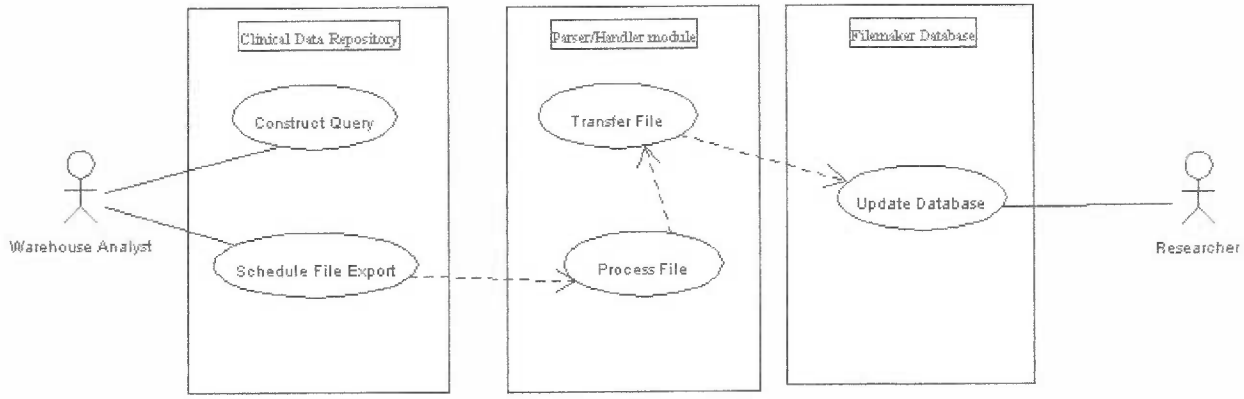


Figure 4 Query construction and file processing

File Handling

Once the clinical datamart was created and transferred to the researcher's FTP account, it becomes necessary to transfer the file to a local directory. A simple shell script written in the Python programming language accomplished this task. Python is a high level, interpreted, object-oriented programming language developed in the early 1990's[9]. It has become very popular with programmers who repeatedly have to write short simple scripts, although it can be scaled to achieve most software tasks. I chose Python for a number of reasons. First of all, as a novice programmer, I found Python relatively easy to understand and implement. It is supported by multiple computing platforms including both PC and Mac. The language itself has robust file handling and text processing

features, which made it ideal for the goals I wished to achieve. Finally, Python has a large following in the open-source community making it easy to find existing well-documented modules that addressed my needs.

Below is the script that was written to transfer the file. As indicated by the code comments (code following the '#' symbol are comments), this script actually does three basic tasks. It first creates a time stamp. It then reads a file from the FTP account, writes it to a file on the researcher's local drive, and appends the file name with the time stamp. Finally, the script sends an email to the researcher to inform him or her that the file now resides in the local directory.

```
#filetransfer.py
#created by Jon Becker 3/8/03
import ftputil
import time
import smtplib
import os.path

date = time.strftime("%m%d%y")
#generates timestamp in mm/dd/yy format

ftp = ftputil.FTPHost("medir.ohsu.edu", "beckerj", "******")
#connects to ftp

path = ftp.getcwd()
#gets the current directory path from ftp

local_path = "C:\Documents and Settings\Jon\Desktop"
#sets a local path where file will be transferred

def filesearch(filename):
#creates a function called "search"

   'searches for a file and transfers its contents'
   #function's docstring, accessed by calling 'search.__doc__'

   if ftp.path.exists(path + filename):
   #checks for file, if file not found then this script does nothing
```

```
        inp = open(path + filename, "r")
        #opens passed file to read

        outp = open(local_path + filename + date + ".txt","w")
        #opens a file with same name and timestamp to write to local directory

        for line in inp.readlines():
            outp.write(line)
            #transfers contents of ftp file to local directory

        ftp.remove(filename)
        #removes file from ftp

        ftp.quit()
        #closes ftp

        inp.close()
        outp.close()
        #closes files

        toaddrs = "beckerjo@ohsu.edu"
        fromaddr = "beckers@ipns.com"
        msg = "File '" + filename + date + ".txt' has been added to your desktop"
        server = smtplib.SMTP('mail.ipns.com')
        server.sendmail(fromaddr, toaddrs, msg)
        server.quit()
        #sends an email message to notify user that the file has been
transferred

filesearch("LEUK_DATA.txt")
filesearch("LEUK_TEXT.txt")
#performs filesearch function for both LCR reports
```

Text Processing

The datamart text files then must be parsed and stored in a format usable by the Filemaker database. While Filemaker is able to import delimited text files, the native LCR reports contain many values that interfere with this process. It is not practical to represent a complete example of these files in this paper's format, but I have included two rows of one of the numeric reports on the following page.

```
01234567@|2003-01-06@|08.55.00@|White Cell Count          @|      5.1000@|5000@|
335883759@|      8378@|      2372@|4127223    @|L  @|    @|    1@|      104@|2003-01-
```

```
06@|10.16.00@|RTIF @|      -@|N @|   5@|- @|C @|      -@|- @|      -@|- @|
-@|P0 @|    1320@|  @|  @|-

012334567@|2003-01-06@|00.00.00@|Hematopathology          @|        -@|8105@|
335883759@|    8436@|    5822@|HBM-03-00029@|L @|  @|  1@|      8@|2003-01-
08@|14.11.00@|RTIF @|      -@|N @|   5@|- @|  @|      -@|- @|      -@|T @|
837@|P0 @|      -@|  @|  @|-
```

Each instance of the characters '@|'marks the end of a data field. This character combination was designated by the LCR analyst on the assumption that it is relatively unlikely for '@|'to occur in any of the data fields. The first field '01234567' is a medical record number. The next two fields contain the date and time that the test was performed. The third field contains the name of the result and the fourth and fifth fields contain the actual test result and the corresponding result code from appendix 1. The remaining fields are essentially useless to the Filemaker database. They contain data points that have not been built and are not required by the users.

The reports for dictated test results follow a similar format. Each row has a patient identifier and temporal information while the last field contains a portion of the result. Therefore, in the following example:

```
01234567@|2003-01-06@|00.00.00@|8105@| 837@|R  @| 33@|Peripheral Blood Smear:  The
red blood cells are present in adequate number

01234567@|2003-01-06@|00.00.00@|8105@| 837@|R  @| 34@|and are normochromic. There
is mild anisocytosis and poikilocytosis
```

The result fields would be joined to read "Peripheral Blood Smear: The red blood cells are present in adequate number and are normochromic. There is mild anisocytosis and poikilocytosis".

The task of processing the text files will also be accomplished using a Python script. It will read each line of text and store the fields as individual strings referred to as 'tokens'. These tokens can then be joined, discarded, or written to a file to be stored until used to update the Filemaker database. By automating the creation of these files, the researchers will have a more complete dataset for analysis by a statistician.

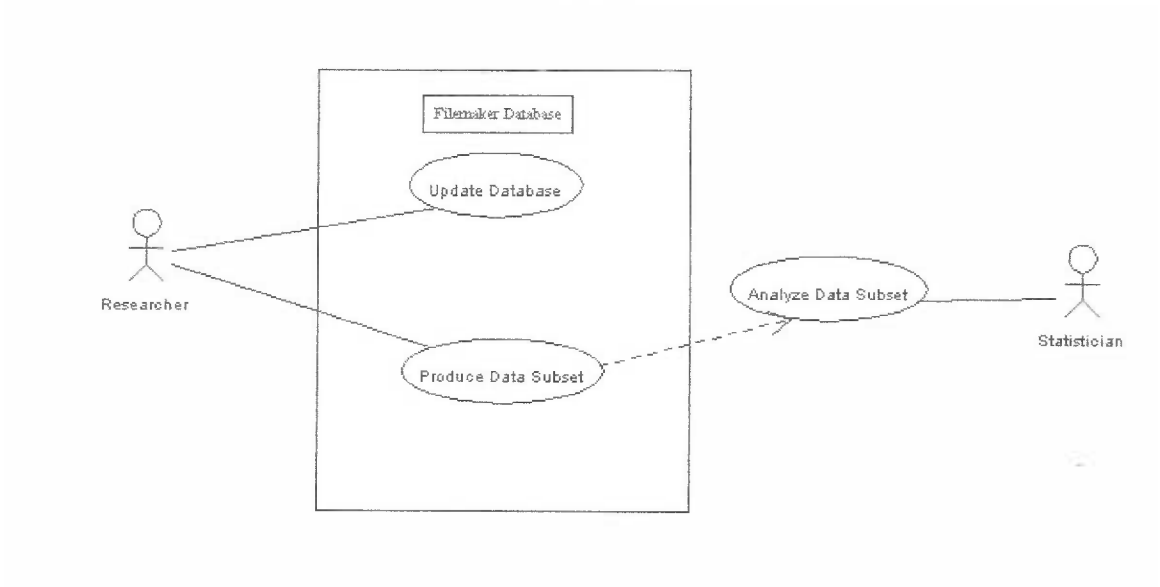


Figure 5 Filemaker use-case

Future Directions

Filemaker Pro was designed for ease of use[19], not for programming an automatic update using a clinical datamart. Although Filemaker has limited support for SQL-92, certain types of statements are not supported (perhaps most notable is the lack a '`create table`' statement). Further, SQL commands can only be accessed through the use of an ODBC driver. The software also lacks support for scheduled tasks, necessitating some form of manual input from a user to complete the update. In order to fully automate this system, this database will have to be implemented in a database management system

other than Filemaker Pro. There are many options currently available that have full support for both SQL and task scheduling.

Metadata Management

In order to develop a model for future LCR datamart projects, it will be necessary to create a metadata repository and management system. *Metadata* is typically defined as data about data. In the context of the data warehouse, metadata refers to information that can be used to effectively support, extract, or exploit the contents of the warehouse[18]. The LCR has many sources of metadata currently in existence:

- Individual departments that contribute to the LCR have records of how data is transmitted (such as the codes in Appendix 1).
- Applications that extract data from the system contain information about the warehouse.
- The vendor for the system has documentation of its structure.
- A small number of people carry a great deal of information about the LCR's structure and content in their brains.

However, most of this information is useless to potential end-users because it is not captured and stored in a central repository[18]. Once such a system is implemented, the burden of providing access to the clinical data warehouse can be taken off the shoulders of over-tasked IT analysts.

The construction of a metadata repository for the LCR will not be a trivial task, however. Since the warehouse was created without a centralized repository for its metadata, this

information will have to be collected from individual sources. This can really only be accomplished by devoting a significant amount of analyst time to the task. It may even be necessary to use only a subset of the LCR's data in the creation of a new data warehouse with a metadata management system.

Types of Metadata

Two basic types of metadata need to be managed as part of the repository: *semantic* and *technical* metadata[17, 18]. Each of these types is primarily distinguished by the tasks that users perform with them. Semantic metadata (sometimes called *business* metadata) refers to information that describes the meaning of the stored data. This description should be understandable by nearly any user and should not require knowledge of database retrieval languages such as SQL. Examples of semantic metadata would include formal data names and definitions, descriptions of terminology, information about structuring queries, and any other data that would help the end user understand the warehouse contents.

Technical or *structural* metadata refers to the physical nature of the data in the warehouse. This includes schema definitions, table architecture, locations of physical storage, user access privileges, and all of the mapping information describing dependencies between individual systems. Researchers wishing to access the LCR would rarely use technical metadata. It is very useful, however, to warehouse analysts and administrators. Technical metadata allows analysts to easily update the warehouse and can be used to track changes and events.

Information about the meaning of data points in the LCR would have been helpful on a number of occasions during this project. One notable example of the need for semantic metadata occurred as we attempted to define temporal aspects of each lab result. To the researcher, the date and time that each patient specimen is collected is of chief importance as they are used to correlate each result with tests from the same patient visit. The warehouse analyst was primarily concerned with the date and time that the result was loaded into the warehouse. This information was used to ensure that each monthly report did not contain results from last month. Both the warehouse analyst and the researcher assumed that the report field OBSV_DATE and OBSV_TIME contained the information they needed. To resolve the issue, individual data points had to be check manually against the departmental systems.

The Complete Package

It is probably unrealistic to design a single "centralized" metadata repository for a large warehouse[10]. While complete access to metadata should be available to all authorized users, it is likely that the metadata will actually be stored in a number of repositories. This fact requires the use of a single standard for how metadata should be structured within the LCR's management system. A number of standards for metadata interoperability are already in existence[18] including the Object Management Group's Common Warehouse Metamodel (CWM). The CWM is an open specification that draws upon the XML, UML, and XMI standard.

Once the necessary metadata was collected and integrated, a graphical user interface could easily be built that would allow authorized users to choose what types of data they

need from the LCR. They could schedule how often they need reports generated and determine how they receive them. Figure 6 illustrates the interaction of all the components of this highly useful system.

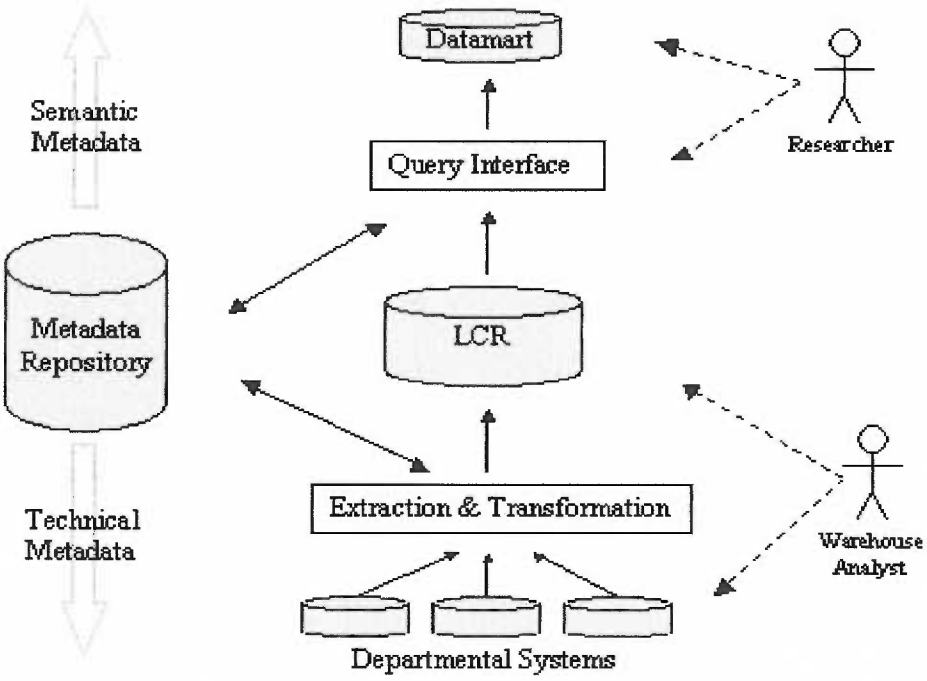


Figure 6 A more usable LCR

Conclusion

Today's clinical data warehouses offer enormous value to healthcare researchers. The creation of subsets of this very reliable data can be used to study large numbers of patients over a long period of time. However, there still exists a great many barriers to the use of this data. Concerns about patient privacy are likely to prevent some research

from ever being conducted[12]. Navigating the political and human resource issues of many healthcare information technology departments can be an exercise in futility (especially as a lowly graduate student).

There is a great deal of work that can be done to improve upon our current system of accessing warehoused clinical data. By undertaking this project I have developed useful skills and gained an understanding of the processes involved with creating and utilizing a data repository. Perhaps the most alarming knowledge I obtained was learning how slowly such a project could progress. By adopting a useful repository and management system for the LCR's metadata, the creation of future datamarts can be accomplished much more rapidly.

Figure1 was created using Staroffice5.2 from Sun Microsystems, figures 3, 4, and 5 were created using Poseidon for UML 1.5.1 from Gentleware AG, and figures 2 and 6 were created using Microsoft's Office Suite.

Bibliography

1. Annas GJ. Medical privacy and medical research—judging the new federal regulations. The New England Journal of Medicine 346(3) 2002: 216-220

2. Bast, R., Kufe, D., Pollock, R., Weichselbaum, R., Holland, J., Frei, E. Cancer Medicine 5th ed. Canada: BC Decker Inc; c2000.

3. Coy KL. The current privacy environment: implications for third-party research. Journal of Continuing Education in the Health Professions 21(2001): 203-214

4. Deshpande AM, Brandt C, Nadkarni PM. Metadata-driven ad hoc query of patient data: meeting the needs of clinical studies. Journal of the American Informatics Association 2002; 9: 362-382

5. Durham Mary L. How research will adapt to HIPAA: a view form within the healthcare delivery system. American Journal of Law and Medicine. 28 (2002): 491-502

6. Evans, Jim. Need for analysis drives data warehouse appeal. Health Management Technology 18:11 (1997) pp28-32.

7. Fox SJ, Wilson RH, Jones JW. Modified HIPAA privacy rule affects research, marketing, security. Managed Care. 2002 Dec; 11(12): 46-7

8. Gunter KP. The HIPAA privacy rule: practical advice for academic and research institutions. Healthcare Financial Management 2002 Feb; pp50-54

9. Hetland ML. Practical Python. Berkeley: Apress; c2002

10. Inmon, W. Building the Data Warehouse (2nd ed.) New York: J. Wiley & Sons c1996

11. McNamee LA, Launsby BD, et al. Scaling an expert system datamart: more facilities in real time. Proc AMIA Annual Symp. 1998

12. Meaney M. Disclosing personal information for research purposes. Case Manager. 2002 Mar-Apr; 13(2): 28-30.

13. Prather JC, Lobach DL, et al. Medical data mining: knowledge discovery in a clinical data warehouse. Proc AMIA Annual Fall Symp. 1997:101-5.

14. Roff, Jason T. UML: a Beginner's Guide. Berkeley: Mcgraw Hill/Osborne; c2003

15. Rogerson CL, Kohlmiller PH, Stutman H. Automated creation of datamarts from a clinical data warehouse, driven by an active metadata repository. Proc AMIA Annual Symp. 1998

16. Shams K., Farishta M. Data warehousing: toward knowledge management. Topics in Health Information Management 21:3 (2001) pp24-32.

17. Stohr T, Muller R, Rahm E. An integrative and uniform model for metadata management in data warehousing environments. Proc. Workshop on Design and Management of Data Warehouses (DMDW'99), Heidelberg, June 1999

18. Vaduva A, Dittrich K. Metadata management for data warehousing: between vision and reality. IEEE Proc 2001 International Database Engineering; Applications Symposium (IDEAS '01), Grenoble, France July 2001:129-135

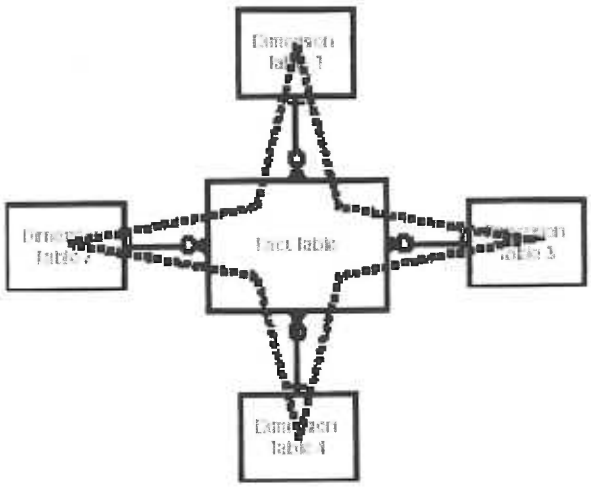
19. Stars, Jonathan. Learn FileMaker Pro 5.5 Plano, Texas: Wordware Publishing c2002

20. Campbell LJ, Patsouris C, Rayeroux KC, Somana K, Januszewicz EH, Szer J.BCR/ABL amplification in chronic myelocytic leukemia blast crisis following imatinib mesylate administration. Cancer Genet Cytogenet. 2002 Nov; 139(1):30-3.

Ultra item	code	Item name
MV	5009	MPV
bA	5140	BASOPHIL %
ba	5138	BASO ABSOLUTE #
eO	5139	EOSINOPHIL %
eo	5137	EOS ABSOLUTE #
hb	5002	HEMOGLOBIN, BLOOD
ht	505	HEMATOCRIT, BLOOD
lY	5130	LYMPHOCYTE %
ly	5133	LYMPHOCYTE ABSOLUTE #
mO	5131	MONOCYTE %
mc	5006	MCHC
mh	5005	MCH
mo	5134	MONOCYTE ABSOLUTE #
mv	5004	MCV
nT	5132	NEUTROPHIL %
ng	5148	RBC FLAG
nt	5135	NEUTROPHIL ABSOLUTE #
pl	565	PLATELET COUNT,BL
rc	5001	RED CELL COUNT, BLOOD
rw	5007	RDW
wc	5000	WHITE CELL COUNT, BLOOD
** the following fields are for textual comments from Ultra.		
item.bf	5293	Diff Comments 1
item.es	5018	Platelet Comment
item.footer.wc	item.footer.wc	
item.mp	5155	Morphology Comment
item.nT	5294	Diff Comments 2
item.rc	5154	RBC Comment
item.xd	5239	Additional Cells

Appendix 1

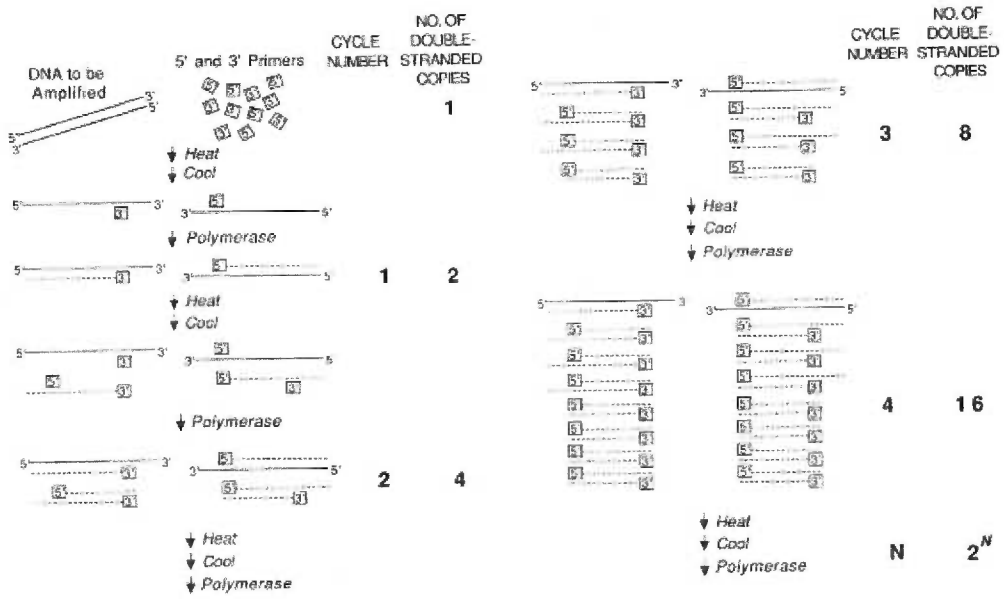
Example of unique identifiers (column headed "code") for CBC results



Appendix 2

Generic star schema for a data warehouse

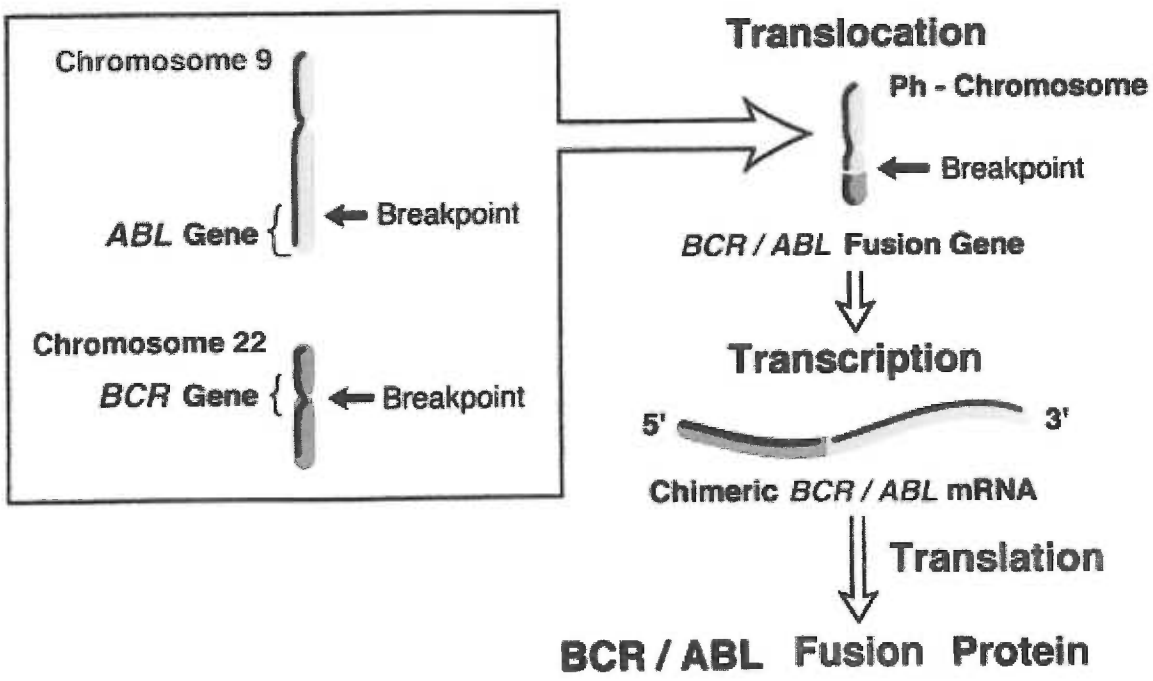
"From Enterprise Models to Dimensional Models: A Methodology for Data Warehouse and Data Mart Design" by Daniel Moody and Mark Kortink which appeared in Proceedings of the International Workshop on Design and Management of Data Warehouses (DMDW'2000) Stockholm, Sweden, June 5-6, 2000



Appendix 3

Polymerase chain reaction (PCR). DNA is mixed with short (10–20 base) single-stranded oligonucleotide primers that are complementary to the 5′ and 3′ ends of the sequence to be amplified. The mixture is heated to dissociate or "melt" all double-stranded DNA, and then cooled to permit the primers to anneal to their complementary sequences on the DNA to be amplified. Note that the 5′ primer will anneal to the "lower" strand, and the 3′ primer will anneal to the "upper" strand. A heat-resistant (thermostable) DNA polymerase was present in the original mixture, and it now synthesizes DNA by starting at the primers and using the strands to which the primers are annealed as a template. This results in the formation of two double-stranded DNA copies for every molecule of double-stranded DNA in the original mixture. The reaction is then heated to melt double-stranded DNA, cooled to allow reannealing, and the polymerase makes new double-stranded DNA again. There are now four double-stranded DNA copies for each original DNA molecule. This process can be repeated *n* times (usually 20–50) to result in 2^n copies of double-stranded DNA.

Cancer Medicine 5th ed. Canada: BC Decker Inc; c2000.



Appendix 4

Summary of the cytogenetic and molecular effects of the Ph Chromosome.

Cancer Medicine 5th ed. Canada: BC Decker Inc; c2000.