

## **Rescuing Data from Decaying and Moribund Clinical Information Systems**

**Jon Patrick, Peng Gao, Xin Li**

*Health Information Technology Research Laboratory  
School of Information Technologies  
University of Sydney, Sydney, NSW, 2006, Australia  
jonpat@it.usyd.edu.au*

### **Abstract**

*This paper outlines a generic methodology to rescuing the data from moribund clinical information systems by reverse engineering the original data model, minimizing the data model for the archival system and migrating the data. The process was developed and tested on five medical systems: three pathology legacy information systems (OMNI-Lab, HOSLAB and HOSREP), a cardiology system (CARDS), and a Breast Screening system. If the data is only required for medico-legal purposes to conform to government regulations then it is not necessary to build any specific interface to the database and access by conventional SQL interface is reasonable. However if it is seen that there is potential to mine the data for research purposes then a more user-friendly interface needs to be provided. We provide a proposal for a Clinical Data Analytics Language (CliniDAL) that serves this purpose.*

### **1 Objectives**

Over several decades many large and complex clinical information systems have evolved to a point where they resist significant further modification and evolution. At such a point these legacy systems have considerable problems, and need to be replaced with an entirely new system but on so doing access to their historical data is usually sacrificed by the organisation. Therefore, there is a need to recover not the application but the essential parts of the data and store it as an archive but in a more contemporary database management system with ready accessibility. This paper aims to identify a viable systematic process to recover the essential parts of the data model of legacy databases and then archive the historical information into a re-conceptualised schema in a more contemporary database management system. The process is tested on a range of information systems to validate its generalisability and appropriateness for clinical information systems.

### **2 Background**

Most hospitals are facing an increasing problem with Clinical Information Systems (CIS) that are being used beyond their use-by date. This can be due to either the need to retain the integrity of the historical data on the system or because the system forms a key part of business processes even though most of the components of the system are no longer used or have never been used.

Over several decades these systems have evolved to a point where they resist significant further modification and evolution. At such a point legacy systems have considerable problems and need to be replaced, however the changeover often leaves the organisation without access to the historical data. The reasons for the migration of legacy data to new software applications are commonly listed as: obsolete hardware and software platforms which are expensive to maintain; tracing and rectifying faults is costly and time consuming due to the lack of documentation and a general lack of understanding of the internal workings of the system; integration efforts are greatly hampered by the absence of clean interfaces, and there is no capacity to expand functionality. Therefore, a viable solution is to gradually abandon the operational processes in the system, as they are taken over by the new system, and reconstitute an archival system that retains the essential parts of the data model from the legacy database and then migrate the historical information into a reconceptualised schema in a more contemporary database management system.

Davies presented an overview of the history and basic principles of Data Reverse Engineering (DRE) [5], Bansleben [2] reviews the literature on data migration, and Bisbal et al [3] review the literature on systems migration proposing a generic migration process of five phases which any approach must address: Justification, Legacy System Understanding, Target System development, Testing, and Migration.

Aebi advocates a data re-engineering process model named MIKADO [1]. The core idea is to introduce an intermediate system to perform some of the difficult tasks before the data is introduced into the target system. This is a practical method to follow especially for a system that needs heavy data cleansing work.

Hammer et al proposed a methodology for Data Reverse Engineering in the SEEK methodology [6]. In this model, the whole process is divided into eight separate steps and the outcome of these steps is produced as an ER Diagram along with the business rules of the application. The eight steps are:

1. AST Generation. This step identifies all the variables in the source code, and then puts them into an Abstract Syntax Tree (AST).
2. Dictionary Extraction. This step identifies the primary keys for each table.
3. Code Analysis. This step works on sub-trees for each variable and tabulates the analysis results, such as constraints implied by source code.
4. Discovering Inclusion Dependencies. This step identifies the dependencies among tables without instructions from system experts.
5. Classification of the Relations. This step labels each relation as strong, regular, weak or specific.
6. Classification of the Attributes. This step classifies each attribute into three categories, (a) Primary Key or Foreign Key, (b) Dangling or General, or (c) Non-Key.
7. Identify Entity Types. This step converts strong/weak relations into strong/weak entities.
8. Identify Relationship Types.

The limitations to these steps have been addressed in Kazi where business needs are identified for the SEEK methodology [7]. The methodology supports construction of supply chain applications, embraces heterogeneity in corporate information systems, provides the ability to extract and compose knowledge resident in sources that vary in the way data is represented and how it can be queried and accessed. The SEEK Project has been used for Knowledge Extraction and Business Knowledge Extraction from Legacy Information Systems [8].

### **3 Target System Development**

The development of an archival system requires firstly the analysis of the legacy system so that it is fully understood. Subsequently a requirements specification for the new system can be prepared. Decisions have to be made with regard to the architecture which should be chosen for the target system. A primary design intention of the architecture in the information systems literature is to facilitate maintenance and extension in the future, so that the developing target system is not the legacy system of the future [3]. However in our case the new system is intended to serve an archival role with limited or no proposal to extend it in the future.

#### **3.1 Testing**

Testing is reported to cost up to eighty percent of a migration engineers' time. The objective is for the target system to ensure it has the same data integrity or better compared to the legacy system. Some testing tools are useful for validation, but experts who are familiar with the legacy systems are the most valuable resource.

#### **3.2 Migration**

The migration phase is concerned with the cut-over from the legacy system to the target system. It must have as little disturbance as possible to the business activities. There are two popular ways to proceed, one strategy is based on gateways [4], and the other is gateway-free, the Butterfly methodology [10]. If the new system is supposed to be an archival database, there is no need to worry about the impact of migration on the current system. The main concern with clinical information systems is how to perform incremental archiving.

### **4 Methods**

The data rescue process we have designed consists of three major stages: Data Model Reverse Engineering

(DMRE), Data Model Minimisation (DMM), and Data Migration (DM). The main purpose of DMRE is to extract a schema from a legacy system. Recreating the schema and relevant documentation of the legacy system provides a better understanding of the system. Typically not all of the functionality available in the Clinical Information System (CIS) has been used and so an analysis of data usage enables culling of unused business processes and data elements. The discovery of the useful business processes and rules implied from the data stored in the legacy repository is essential. After applying DMRE, the goal of DMM is to construct a functionally equivalent data model, and translate it into a database schema on a new database management system (DBMS). DM involves the loading of the culled and re-organised data into the data model.

Data Model Reverse Engineering is defined as the application of analytical techniques to one or more legacy data sources to elicit structural information (such as term definitions and schema definitions) from the legacy sources in order to improve the database design or produce missing schema documentation. DMRE applied to this project includes the following five processes:

1. Data Dictionary Extraction.
2. Classification of the Entities and Attributes.
3. Discovering the Relationships and Dependencies.
4. Generating an Entity Relationship Model (ERM) for the legacy system.
5. Discovering the Business Processes and Rules.

Figure 1 illustrates the processes and methodology of DMRE applied to the OMNI-Lab system. The following sections introduce these five phases and related activities in detail.

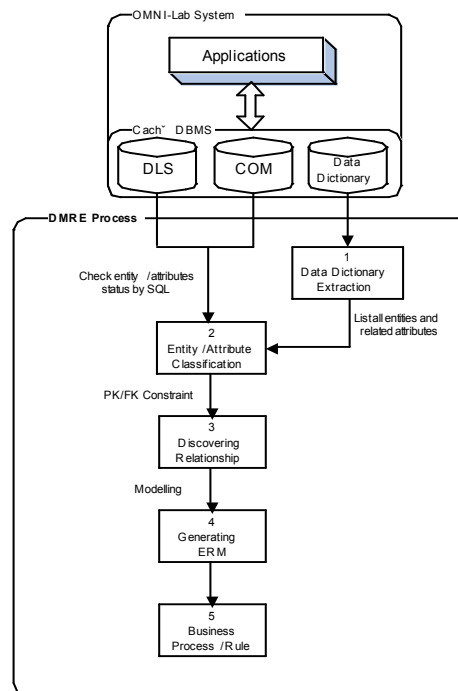


Figure 1: DMRE Processes applied to OMNI-Lab System

#### 4.1 The Procedure of DMM and DM

Data Model Minimisation (DMM) requires that the schema in the archival system should be as simple as possible but with assured information consistency. The main purpose is to provide a simple schema for a novice database analyst to understand and retrieve the information by joining as few entities as possible in the archival database with an appropriately well documented design.

A DMM approach includes the following processes:

- Constructing the new archival Entity Relationship Model: This work involves identifying the parts of the original model that are no longer needed, because they have never been in use, they are part of the operational needs (e.g. billing), or they are superseded by later technology (e.g. hardware descriptions)
- Creating the archival Schema in MySQL: This is the process of using a MYSQL interface to create the physical database matching the newly designed schema.

Data Migration includes the steps of:

- Exporting raw data from the legacy system: Requires reading data from the legacy database and placing it in temporary files.
- Data Cleaning: is the process of viewing the data records and checking if data values are legal for the data type. This uses programs written to check for valid data types and values.
- Importing the Data into the MySQL archival data warehouse: This is a process where the records are re-assembled to match the new data model. This is not a simple read of the records into a database. Commonly the data has been reorganised so the records of different types have to be brought together to make composite records spanning different tables to their configuration in the legacy system. Figure 2 illustrates an overview of the process of DMM and DM after DMRE.

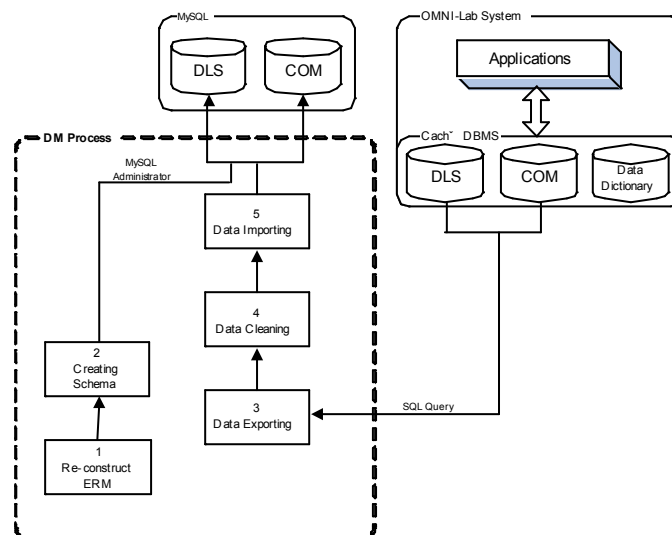


Figure 2: Data Migration Processes applied to a MySQL Archival system.

## 5 Results

The new approach or generic methodology has resulted in a significant reduction of the number of attributes and the number of tables in each system. Particular difficulties in identifying the nature of the data inter-relationships are described and issues in data veracity and data cleaning are assessed.

The first four systems originated from hospital based services in the state of New South Wales, Australia. The OMNILAB and HOSLAB systems were in use by the South Eastern Area Laboratory Services being an operational department of the South East Sydney and Illawara Area Health Service. OMNILAB consisted of anatomical pathology content while HOSLAB was used to manage all blood services. CARDS was a system used in the Cardiology department of the Westmead Hospital, Sydney to track patient care whilst they attended hospital. HOSREP was a database management system in use the South West Pathology Services of the Sydney South West Area Health Service based at the Liverpool Hospital to record all anatomical pathology cases. The Breast Screening system was managed by a service set up by NSW Health to operate breast screening across the state and is used to record and track patients period in usage of the service and the

results of the screenings.

**Table 1. Comparison of the sizes of five databases before and after their rescue.**

<b>SYSTEM</b>	<b>Original Tables</b>	<b>Original Attributes</b>	<b>Reduction in Tables</b>	<b>Reduction in Attributes</b>	<b>Change in Data Records</b>
OMNILAB (SEALS)	79	794	35.4%	22.3%	undetermined
CARDS (SWAHS)	255	7988	79.2%	48.6%	-39.1%
HOS-LAB (SEALS)	21	136	66.7%	43.4	undetermined
HOSREP (SWAPS)	21	172	38.2%	68.6%	+4%
Breast Screening (NSWBCI)	15	801	11.8%	22.2%	-4%

Assessment of the reconstituted databases has been performed by designing up to 20 queries that cover a range of the materials included in each database. The queries were executed over the original database and then over the redesigned databases. In each case we were able to demonstrate that the same answers were provided to the test queries, thus verifying to a reasonable degree that each reconstituted database was consistent with each original database. The testing did not attempt to verify every attribute in every table.

## 6 Discussion

The process developed over five data rescue projects has proven to be efficient and reliable. With each new legacy system it has been honed as well as expanded to deal with different CIS and their underlying database management systems. The lengthiest part of the task has moved from the data model reverse engineering to the data migration stage as larger systems have been tackled, and our expertise improved.

With a greater understanding of the range of issues that emerge from data rescue cases we are building generic tools especially for the data migration stage where data has to be cleaned and validated before transfer into the archival system.

This paper has not addressed the issue of user interfaces in archival systems. Generally users have been grateful for rescuing the data and having it in a form where they can execute SQL statements over the database directly. This requires a reliable data model to be available to new staff who will design and execute the queries. The data is placed into an open source database management system to avoid lock-in to proprietary software given that the aim of the new system is to escape an old locked-in relationship. Nevertheless a basic user interface that permits easier construction of queries over fundamental variables is desirable.

It would be more desirable to provide a customized Graphical User Interface to make it easy to produce ad hoc queries of the database. There is no value in producing a reporting system as the database is no longer operational and so all the data in is static. It would be possible to provide a standard report generating tool like Crystal Objects to provide for standard reports but this would not readily support ad hoc queries.

A view of the database as a repository for data analytics to support retrospective research entails a different perspective to the requirements of the processing interface. In these circumstances a more satisfactory solution would be a mechanism to make entirely as hoc queries of the database with the ability to pursue a line of research investigation. We have designed a general purpose clinical data analytics language (CliniDAL) as an enhancement technology for clinical information systems. A version of the CliniDAL has been modified to fit onto the HOSREP database of SWAPS. It supports queries over the contents of the structured parts of the database but also the anatomical pathology text reports. The text reports have been indexed using another

technology developed in our laboratory which is able to detect the SNOMED CT code equivalents to the texts [9]. It accepts references to the major categories of information in SCT including demographic details, diagnosis codes, diagnosis text as SNOMED CT concept descriptions, and time spans. Installing the CliniDAL software on the SWAPS database was part of our research program to demonstrate the portability of CliniDAL but it now needs to be used to pursue research questions of interest to demonstrate its effectiveness.

## 7 Conclusions

The process developed to rescue data from moribund clinical information systems has proved durable by its trials on 5 different types of systems. The process has been tested and honed with every reuse. Now it remains to determine efficient methods for generating user interfaces that are easy to use but do not obstruct the users' access to the underlying data and data model so as to create unfettered access to the data. Progress has been made in this quarter by installing a Clinical Data Analytics Language on the SWAPS database of anatomical pathology reports, which provides access to the contents of free text in the query structure.

## Acknowledgements

Students of the School of IT were involved in the developments of these projects: Peng Gao (CARDS), Xin Li (OMNILAB), Hui Ke(HOSLAB), Kiran Abraham(HOSREP), **Error! Reference source not found.** (Breast Screening) and their contribution is greatly appreciated. Likewise, the CliniDAL system was installed on the SWAPS data warehouse by Victor Yingze Zhou and we appreciate his contribution.

## References

- [1] Aebi D. "Data Re-Engineering - A Case Study". Proceedings of the *First East-European Symposium on Advances in Databases and Information Systems (ADBIS'97)*, pp. 305-310, 1997.
- [2] Bansleben, EP, Haas. SW, "Database Migration: A Literature Review and Case Study. A Master's paper for the M.S. in I.S. degree", School of Information and Library Science, University of North Carolina, USA, November, 2004.
- [3] Bisbal J, Lawless D, Wu B, Grimson J. "Legacy Information System Migration: A Brief Review of Problems, Solutions and Research Issues". Available at <http://www.cs.uu.nl/docs/vakken/swa/20012002/Reader/07-TCD-CS-1999-38.pdf> on 1/4/07.
- [4] Brodie R. *Migrating Legacy Systems: Gateways, Interfaces and the Incremental Approach*. Morgan Kaufmann Publishers, Inc. USA, 1995.
- [5] K. H. Davis, P. Aiken. *Data Reverse Engineering: A Historical Survey*. Proceedings of IEEE Seventh Working Conference on Reverse Engineering, November 2000, pp. 70-78.
- [6] Hammer J, Schmalz M, O'Brien W, Shekar S, Haldevnekar N. "SEEKING Knowledge in Legacy Information Systems to Support Interoperability". Proceedings of the International Workshop on Ontologies and Semantic Interoperability (ECAI-02), Lyon, France, July 2002, pp. 67-74.
- [7] Kazi AS, O'Brien WJ, Issa RRA, Hammer J, Schmalz MS, Geunes J, Bai SX. "SEEK: Accomplishing Enterprise Information Integration across Heterogeneous Sources". Published at <http://www.itcon.org/2002/7> in August 2002.
- [8] Paradauskas B, Laurikaitis A. "Business Knowledge Extraction from Legacy Information Systems". *INFORMATION TECHNOLOGY AND CONTROL*, 2006, Vol.35, No.3, pp 214-221.
- [9] Patrick, J, Budd, P. & Wang, Y. &. An automated system for Conversion of Clinical notes into SNOMED CT. *Health Knowledge & Data Mining Workshop, Ballarat, Research & Practice in Information Technology*, 68, 195-202, 2007.
- [10] B. Wu, D. Lawless, J. Bisbal, J. Grimson, Vincent Wade, R. Richardson, D. O'Sullivan. "The Butterfly Methodology: A Gateway-free Approach for Migrating Legacy Information Systems". Proceedings of the 3rd IEEE Conference on Engineering of Complex Computer Systems (ICECCS '97), Villa Olmo, Como, Italy, pp. 200.