Chapter VII
Transforming Legacy Systems into a Valuable Heritage: The Case of the FixIT Migration Tools in Finland

Mikko Korpela, Juha Mykkänen, Mika Räsänen, Hellevi Ruonamaa, and Marko Sormunen
University of Kuopio

A migration strategy can turn legacy systems into a valuable heritage. In Finland, most hospital information systems are based on the U.S. Department of Veterans Affairs FileMan/Kernel technology. By the mid-1990s, the user interface and system architecture had become obsolete. The University of Kuopio together with a consortium of vendors and hospitals developed a migration strategy and toolkit, Delphi-FixIT, to move the viable systems to the client/server architecture and graphic user interface. An alternative toolkit, Web-FixIT, was developed in 1998–99 to allow a further migration to the Web browser interface and Java applets. The next step in 2000–01 will be to encapsulate existing functionality into components. This will make it possible to replace even the FileMan database by an object database at some later time. The migration strategy will thus keep the systems always abreast with technological progress without major disruptions. The technological solution developed is argued to be scalable down to very small setups in Nigeria, and up to quite large setups in the USA.

INTRODUCTION

Large operational information systems, e.g., hospital information systems (HIS), tend to be surprisingly long-lived, far beyond the expectations of their original developers. If care is not taken, such systems become technologically and functionally outdated legacy systems. For instance in Finland, the most popular applications packages for patient information management in hospitals, the Musti family of packages, were developed some 15 years ago, and the Finstar package for health centers (primary healthcare) even before. Both of them are based on dumb terminal access to mainframe computers. Big university hospitals have calculated that it will take four to six more years to completely replace the...
terminal-based technology and move to client/server applications. In the USA, the hospital information systems of the Department of Veterans Affairs (VA) and the Department of Defense (DoD) are based on the same software technology as the Musti systems.

In this chapter we present a migration strategy and technology which was developed to move the viable Musti applications first to the Windows-based (“fat”) client/server architecture and graphic user interfaces, and further to the Web browser-based (“thin”) architecture and interface. The next steps and future challenges on the migration path are presented, and the wider applicability of the technology discussed. The objective is to describe the strategy and migration technology as well as their development processes in sufficient detail to allow systems developers and technology managers with other kinds of legacy technologies to adopt ideas for their own migration solutions.

BACKGROUND: LIFECYCLE MANAGEMENT OF HOSPITAL INFORMATION SYSTEMS

The overall lifecycle of large hospital information systems can be managed by basically two different long-term strategies. Firstly, one can try to freeze the HIS as much as possible until it has become obsolete, and then replace everything at one time. The second option is to design a migration strategy and apply stepwise renovation. With the latter option, the idea is to design, implement and update a plan for ”modernizing” different parts and aspects of the overall HIS in a piecemeal manner and before any old part or aspect becomes a severe obstacle to further development.

More specifically, the following aspects of an HIS application package can be considered for renovation, with some degree of independence from each other:

- Functionality (the support provided to clinical and management purposes)
- User interface
- Internal software structure of the application
- Database
- Hardware platform
- Network infrastructure
- Application interfaces (the linkages between subsystems as well as interorganizational linkages)

It may be advisable to try to minimize the number of aspects that are thoroughly changed at the same time. For instance, if a given subsystem (application package) is not serving changing clinical purposes well any more, it may be better to freeze the user interface, software architecture and application interface technologies while redesigning the functionality and the database contents. Likewise, while the user interface technology is being thoroughly changed, it makes life easier for the users if the functionality and data contents are retained familiar. If most or all aspects of a system need to be changed at the same time, it is likely that it does not provide a sufficient basis for renovation, and it is cheaper and easier to replace it by a completely new one.

In the rest of this chapter, we illustrate the HIS renovation principles by presenting and discussing a case in which a migration strategy and technology were developed for a specific class of legacy systems in Finland. The user interface and software architecture were considered the most urgent aspects for renovation in this case, but the strategy and technology provide a path for gradually replacing all parts and aspects of the systems in a
Communication Assessment Checklist in Health: Assessment and Comparison of Web-Based Health Resources


www.igi-global.com/chapter/communication-assessment-checklist-in-health/192701?camid=4v1a