

Volume 2 / Issue 3 2007 - Cover Story

The Legacy Systems Challenge

Over the past decades, European e-Health projects have principally focused on research and development.

Some have also been demonstrated on a large scale through programmes like eTEN. However, many are now close to real-life deployment and use, and the European Union's e-Health Action Plan has become a central plank of its broader i2010 strategy.

According to interviews and briefings conducted by HITM with healthcare IT managers in a variety of EU countries, legacy systems may pose challenges for the European Union's e-Health vision to become a reality.

A robust European e-Health infrastructure will need to distribute and deliver personalised healthcare on demand, and do so in real-time.

Two factors will determine whether this is achieved, sustainably and efficiently.

First, all healthcare IT systems must be able to communicate with each other, and do so all the time. Second, they should also interact seamlessly with new devices – such as wireless transmission, speech-recognition and radio frequency identification (RFID), portable input/access terminals and mobile lab units – as well as emerging technologies like patient biomarkers and smart devices which are directed at enabling home-based, on-demand care.

Disparate Systems and Infrastructural Bottlenecks

Such a scenario is, however, not straightforward. Even now, it is common to find different departments of hospitals having their own IT systems and databases. As a result, internal communications can still be confined to paper printouts - and some physicians, at least, seem to prefer it that way. More problematic is the fact that small differences in spelling or address make production of a single electronic record for each patient or customer immensely time-consuming.

These bottlenecks must be sorted out to meaningfully tackle issues of convergence and seamless interaction with emerging – and future - e-Health era technologies. The reason: the back-end of the EU healthcare infrastructure is evidently crucial for an end-to-end e-Health solution. However, such a back-end is often based on proprietary legacy systems, and then again, not rarely, in the shape of relatively-elderly mainframe computers.

Although robust, legacy systems are limited in terms of modern functionalities and connectivity. They were designed for an era when processing power was a premium product rather than the commodity it has since become. They operate within what the IT industry terms isolated "silos". While a variety of technological workarounds (above all in the shape of middleware) have build gateways into such silos, very few can predict whether they will cope with the explosion of realtime data access/transfer requirements of a distributed, realtime e-Health environment.

Users have also been occasionally hit by withdrawal of support for platforms considered non-profitable, or squeezed by escalating license and support costs. Some have also been caught on the wrong side of a paradigm shift in technology. This was the case with minicomputers in the early 1990s. A more recent case was Hewlett-Packard's decision in 2002 to drop its popular 3000 series, after its proprietary MPE operating system began losing market share to rivals. Caught unawares were 2,000-odd users in Europe (including several hospitals). Some had installed 'new generation' HP e3000s – enticingly upgraded with new processors - barely months previously.

Of Familiarity and Comfort Factors

So far, responses to legacy IT challenges have been ad-hoc. In several instances, they are driven by vendors and the momentum of technology, rather than the needs of users. Nor have most of the responses involved a realistic assessment about the scale and expected impact of underlying legacy IT problems, or even sought to learn from the experience of other healthcare IT manager peers.

Indeed, despite their shortcomings, several facilities have found compelling reasons to maintain their legacy IT systems.

Legacy IT systems represent a significant investment. Moreover, they are robust and deliver – and do this at least for present-day needs; these, of course, remain the highest priority for hard-pressed IT managers. The incentive for inaction is highlighted further when we consider that in Europe, most healthcare IT managers are considered (and view themselves as) risk-avoiders. Although some IT users see replacement of legacy systems bringing platform- and vendor- independence, many remain apprehensive about scalability, and above all, system stability.

Legacy systems often house mission-critical information, which requires near-100% availability. This means that the IT system cannot be taken out of service for modernisation.

Meanwhile, the cost of designing a modern IT system with similar availability levels is seen to be (sometimes-prohibitively) high, especially when managers have to struggle for budgets in an era of organisational consolidation and financial pressure. Such factors can have serious consequences. For example, a study published in the Journal of the American Medical Association ('Physician Order Entry Systems in Facilitating Medication Errors'. JAMA. 293,10: 1197-1203) reported how a computerised practitioner order entry (CPOE) system installed at an academic hospital resulted in no fewer than 22 types of medication errors, due to poor system design.

IT managers also see their legacy application as the result of decades of progressive refinement, and the accumulated wisdom of a generation of workers about the rules that run their organisation. Adding to barriers to change here is the fact that refinements are usually undocumented. Some are believed to be critical – and cannot be easily deciphered or replicated.

Another major hurdle is cultural. Changeovers face resistance as they require redrawing of turf and relative positions within internal hospital staff hierarchies. In a commentary for the Spring 2007 issue of Healthcare IT Management, the CIO of France's Arras Hospital noted that a legacy systems overhaul at his hospital was "a major undertaking in its own right" and did not seem viable "without a parallel re-structuring of the organisation's internal processes and a change in its culture."

Vendor relationships, too, encourage the status quo. As a 2004 study by market researchers Frost & Sullivan determined, established vendors tend to have a distinct advantage, due to strong relationships with hospitals which continue to depend on legacy systems. Most hospital IT departments, furthermore, believe they do not have the funds required for modernising legacy systems, according to the study.

Ways Out, and Around

One of the instinctively-favoured solutions for legacy IT problems, to re-engineer the system in a Big Bang modernization, has begun to show evidence of serious limitations in large-scale environments like healthcare.

In the UK, for example, the ambitious multibillion pound program to modernise the IT infrastructure of the National Health Service (billed as the world's largest civilian IT project), has run into serious difficulties; this has been discussed in a previous issue of Healthcare IT Management. More recently, modernising two legacy systems at Britain's Barnet and Chase Farm Hospitals trust saw patients receive wrong appointment letters. According to Alex Nunes, a patient representative, the Trust had been sending appointment letters "to people who haven't got appointments and for people who should have appointments it hasn't." Mr. Nunes said he saw someone really shaken by being wrongly called for major surgery. Misdirected letters, he added, could result in people not getting the treatment they actually need.

Some IT managers have sought solutions which do not necessitate replacement of legacy systems. They have instead moved on a hybrid route - keeping the procedural component of the legacy system in a source code format while migrating data, or by replicating functionality in core systems with downstream systems or interfaces that bypass primary processing rules, among them GUI (graphic user interfaces), datawarehousing and a variety of bespoke systems – such as rewriting part of the legacy code for new platforms.

The outcome is again not always desirable: two incompatible cultures within the hospital IT staff – one in modern ehealth/ e-business and Web initiatives, and the other, in existing legacy system administration. This problem is compounded because of the sometimes uncoordinated nature of transitional measures with what one IT manager observed was a "nightmarish" mix of isolated client-server and customized packaged applications.

A final threat is the expected decline in the availability of legacy IT skills. Even as the bulk of legacy programmers in Europe approach retirement age, legacy languages are no longer on the curricula of IT training institutes. Recently, according to one of our interviewees, there have been moves in the UK to reintroduce legacy IT language courses, especially Cobol, at Oxford Brookes University and the University of Central England.

Beyond Healthcare: The Legacy 'Ticking Time Bomb'

Legacy challenges apply to other sectors, too. Certain banks in the European Single European Payments Area (SEPA) have found their legacy systems cannot accommodate extra information such as IBAN (International Bank Account Number) and BIC (Bank Identifier Code), and a

senior EU Commission official has publicly warned that some may well miss the SEPA deadline “and even also fail to stay in business.”

In the wider industrial/corporate area, a survey in summer 2006 by Britain’s National Computing Center concluded that legacy IT systems were a ‘ticking time bomb’. Four out of 5 of the companies in the survey reported that a lack of (legacy) system agility made it difficult to align IT with business objectives, and over 60 per cent believed there was a negative return on investment (RoI) in the maintenance of legacy systems.

The demands of e-Health will be at least as strenuous as those facing banks and industrial corporations. Clearly therefore the key to building a dynamic and viable e-Health infrastructure will depend on quantifying, understanding and finding solutions for the challenges accompanying hospital legacy IT systems.

Technical Choices for Legacy IT Problems

A final threat is the expected decline in the availability of legacy IT skills. Even as the bulk of legacy programmers in Europe approach retirement age, legacy languages are no longer on the curricula of IT training institutes. Recently, according to one of our interviewees, there have been moves in the UK to reintroduce legacy IT language courses, especially Cobol, at Oxford Brookes University and the University of Central England.

Gateways

One of the simplest workarounds is to wrap legacy data in gateway software such as Sun Microsystems’ Java Database Connectivity (JDBC) or Microsoft’s Open Database Connectivity (ODBC). Some users seek to wrap not just data but also the business rules of legacy systems through the Common Gateway Interface (CGI) - a bridge between external applications, information servers and legacy systems.

RE-Engineering Applications

This involves the rewriting of applications from scratch, using the latest tools and technologies, converting all data structures and subsequently the data itself, along with its business logic. In an ideal world, where applications are small and there is ample staff and money, the re-engineering option is attractive. Even in some large-scale projects, where original application code has run out of control over decades, this is sometimes the clearly preferred option. However, for the large majority of applications, reengineering is the most expensive and least successful route. Understanding complex business rules and algorithms contained within a system, especially without access to the original programmers and system architects, is a tough call at the best of times. In the real world, yet another problem is that requirements are themselves changing and may have mutated into something new by the time the re-engineered application is ready to go live.

Package Implementation

For generic applications, a popular option is to integrate a package like SAP, Oracle etc. Indeed, many legacy applications are considered to be generic - for example, accounting, manufacturing, human resources etc. In addition, since package vendors have a large customer base with millions of dollars invested, they have sought to stay compatible with customer legacy IT architectures.

As a result, they have not completely redesigned applications but instead wrapped existing code to be invoked by industry-standard component models such as CORBA (common object request broker architecture) and COM (Component Object Model). Package implementation also confronts other problems. They require modifying of a user’s specific business practices. This is usually entrusted to in-house programmers, who make the necessary changes.

However, follow-on changes are once again entrusted to the in-house team, rather than a vendor, for all new releases of the package. As a result, hospital IT staff end up with increasing maintenance responsibility.

Meanwhile, those who have chosen packages, but decide not to upgrade to new releases (or versions), face another major risk – withdrawal of support by the vendor for older versions. Oracle, for example, has stopped technical support for all its Version 10 applications.

Middleware and EAI

Middleware or Enterprise Application Integration (EAI) tools go one step further. They employ methodologies such as object- and component-orientation; they are run on Web servers which utilize object architecture like CORBA, EJB (Enterprise Java Beans) or COM. Like CGI, middleware/EAI also faces a steep increase in (permutational and combinational) complexity as new devices are incorporated into the IT system. This is expected to be a result of e-Health and other previous infrastructural modernization programs; their scale is expected to only grow with time.

In an analysis of EAI, the 'Economist' warned : EAI "is expensive, and so far it connects only a limited set of applications. But its biggest disadvantage ... is that it does not provide the flexibility of changing business processes in real time."

Migration/Translation

Migration involves the transfer and redeployment of current programs on to alternative, less expensive hardware platforms. However, there are often significant hidden costs. Migration also rarely provides extra functionality, which is not seen as its essential objective.

A related option is to translate existing legacy IT applications to current-generation programming technologies such as Java/J2EE or Visual Basic/DotNet. A strong argument for migration and translation is not the preservation of data for its own sake, but the preservation of business rules, data rules, process flow, computing resources and application integrity.

XML

XML (extensible mark-up language) is aimed at allowing exchange of information between disparate systems. However, the global IT industry has yet to agree to a set standard for data and field definitions, and there is a continuing lack of browser support and end user applications. It is also becoming likely that only newer IT systems would be designed to comprehend XML; this raises questions about the time and costs required to convert existing legacy databases to XML. In brief, XML is still experimental and not solidified.

Published on : Mon, 31 Dec 2007