

The Future of Information Technology: Selecting a Forward-Looking LIMS Vendor

by David Leitham

Most sophisticated LIMS purchasers have a well-developed methodology for evaluating which system best meets their needs. Current requirements are carefully documented, weighted, and compared to the functionality provided by commercially available solutions. However, given the level of investment made in LIMS and their mission-critical nature, the long-term viability of the vendor must also be considered. This goes beyond a simple analysis of the vendor's fiscal health; it is becoming increasingly important to look at how well the vendor is positioned to meet the client's future needs. Unfortunately, evaluating whether a vendor can continue to meet one's business needs over time and future-proof one's investment is a less precise science.

Crossroads for LIMS and software

Since its early days, the development of complex software has been a struggle. The term "software crisis," used to sum up these efforts, has been in existence almost as long as the term "software engineering." Over the years, software developers have adopted varying methodologies, ranging from basic Waterfall to Extreme Programming (XP), to address these failings. While many of these processes have helped to improve initial software quality, they often fell short in addressing the issues of changing complex software. Today's software crisis refers primarily to the software developer's challenge to keep up with the rate of change of business.

Historically, from a software perspective, LIMS have simply been highly

configurable data storage systems. To meet the market demands of the time, which called for the systems to meet each customer's individual processes, LIMS vendors developed little in their core product and left much of the functionality to customizations that were needed to meet individual requirements. While this approach met the current need, it had significant drawbacks. Perhaps most noteworthy was the difficulty upgrading to new versions when the original is highly customized. Due to the bother involved, LIMS customers were very

In an era of innovation and rapid change, vendors will be differentiated by their ability to reliably adapt and modify their products.

conservative with implementing upgrades. This situation contributed to an understandable stagnation in development and dearth of new technology in the LIMS industry.

However, a disruption in this status quo is underway. Increasingly, LIMS customers have learned that highly customized software is problematic and are willing to adopt more standardized, commercial off-the-shelf products. Initially, this might not seem like such a challenge for vendors

since many can simply offer a standardized version of their current offering. However, one of the key outgrowths of standardization will be that upgrades, and the ability to accept new innovations, will become considerably easier than they are now. Without customizations, new software can simply be installed. Validation times should be reduced appreciably as well, since standardized products can yield standardized test scripts. With these issues addressed, the LIMS field will begin to resemble other arenas of information technology, in which innovation and change are the norm. Increasingly, LIMS vendors will be evaluated based on their ability to advance and keep pace with more rapidly changing needs.

This will challenge the way in which many traditional LIMS vendors develop software. The old methods and technology may have been sufficient at one time, but they cannot necessarily survive in a more aggressive environment. In an era of innovation and rapid change, vendors will be differentiated by their ability to reliably adapt and modify their products. This ability will be shaped by the decisions they make today.

Indeed, standardization may do to many LIMS vendors what a meteor is theorized to have done to equally prehistoric beasts.

The question, then, is how to weed out these outdated products. Progressive LIMS developers should continually strive to create a new and innovative product. While this article does not provide a comprehensive list of evaluation criteria, it offers useful suggestions to foster discussion with potential vendors. The vendor should

anticipate these issues and be able to articulate its approach to each.

Technology plan

It is the responsibility of the vendor to contemplate the future of information technology so that the purchaser does not have to. The vendor should be able to clearly outline its plans for adopting new technology, for example .NET (Microsoft, Redmond, WA) or Java® platforms (Sun Microsystems, Santa Clara, CA). This is not intended to advocate one technology over another, particularly since there is no one approach; however, the vendor must have a plan for keeping one's information technology current. This plan should take into account what assets are already present (legacy, or cherished code), keeping in mind the end goal and a logical roadmap to reach it. The vendor should also be able to explain the business benefits of the targeted new technology.

The importance of a technology plan traces back to keeping up with the rate of change. While it is true that virtually any system can be built with practically any programming platform, each new generation of programming platforms has brought about increased developer productivity and an improvement in addressing the effort of changing complex software. A vendor that remains rooted in older technologies will be increasingly hampered to keep up with those that adopt new technology.

This is not to say that vendors should jump at every new technology. Vendors should, however, continuously evaluate new technologies. Adopting technologies can be intensive for both the vendor and customer; therefore, common sense must be employed. There should be a clear business benefit to the customer to make such a move. When that benefit outweighs the cost of transition, it should become part of the technology plan. It is important for the purchaser to ask the following questions: Can the vendor explain why one technology path was chosen over another? Was the decision

made based on pragmatic analysis? Such insight may color future decisions with regard to that vendor.

Since information technology is constantly evolving, the technology plan must be a living entity. Therefore, beyond simply articulating the plan, the vendor should have a clear owner of the plan and the facilities in place to continuously evaluate new and changing technologies for potential impact on the plan.

Thinking beyond today's requirements

Good software developers do more than just develop software based strictly on the current set of requirements. Recognizing that changing

While it is preferable to choose a vendor with a promising future, there will not be much business benefit unless that vendor can bring its customers along.

complex software has been the bane of the industry, they must anticipate ways in which the software may change over time and build in that flexibility from the start. Such flexibility is derived not only from the design of the software, but from the very processes used to develop the software as well.

From a design perspective, there are numerous time-tested best practices, such as separation of business logic, user interface, and data access code within a system, to the latest thinking in software architectures. Much like with technology, vendors cannot simply adopt every new approach, but they should monitor and evaluate the latest trends to incorporate into their own practices.

One design trend that a vendor should be aware of is an architectural paradigm called Service Oriented Architecture (SOA). In this approach, applications are "built" by assembling loosely coupled services, each of which provides a unit of business logic. This method represents the latest in the evolution of distributed computing, which has included technologies such as Remote Procedure Call (RPC), Common Object Request Broker Architecture (CORBA), and Distributed Common Object Model (DCOM). SOA improves upon these earlier approaches due to its stronger emphasis on standards and loose coupling between services. SOA has received strong support from major software vendors, is governed by the World Wide Web Consortium (W3C), and is advocated by both Sun Microsystems and Microsoft.

SOA is not only an incremental improvement in the way complex software is built, but could prove to have a profound impact on the way software is delivered and deployed. The promise of SOA is that in the not too distant future vendors will deliver services that can be easily integrated with other services in one's enterprise, giving one greater ability to assemble precisely the functionality needed with standardized services. This also opens the way to selectively upgrading portions of one's applications to obtain new features with reduced impact and revalidation on other services.

Given the scope and widespread adoption of this approach, the vendor should at the very least be able to convey its view on the subject. Ideally, the vendor already has plans in place to either adopt SOA design practices or provide a service-oriented interface to its offerings.

The ends are a product of the means

Perhaps even more important than the specific technology used to develop a product is the software development process that is used. Sadly, innovations in the processes used to develop soft-

ware are often overlooked by developers in favor of slick new development tools. Much like development tools, though, there has been significant growth in this area, and the vendor should be committed to continual process improvement.

While the established, step-wise waterfall approach has its virtues, it is particularly weak at adapting to changing needs. In response, more and more iterative development techniques have surfaced. One of the more promising has been termed “agile development.” In this case, “agile” refers to maneuverability, not necessarily speed, though increased speed is a common by-product. Agile development groups together a set of software development practices that, in general, apply time-boxed iterations, adaptive planning, and evolutionary delivery. What sets the approach apart from other iterative techniques is its strong emphasis on continuous integration and unit testing. In the agile world, before a unit of code is written, code is first written to test that unit. While it sounds laborious at

first, it ensures future maneuverability. It allows one to quickly adapt an aspect of one’s system to changing requirements and immediately determine that change’s impact on the rest of the system by exercising all of these unit tests.

As with technology, a one-process-fits-all approach is not being promoted here. The vendor, however, should not be content that its current processes will forever allow it to keep pace with the rate of business change. The vendor should be able to explain the latest software trends and have a record that indicates process review and improvement.

Conclusion

While it is preferable to choose a vendor with a promising future, there will not be much business benefit unless that vendor can bring its customers along. This starts with the effort it will take to upgrade from one’s current platform. While a vendor may be able to speak to a great vision with bold new functionality, if its current offer-

ing requires extensive customization to meet the client’s needs, it is unlikely that vision will be realized any time soon.

Seeing the increased demand for standardized LIMS, the vendor should already be moving to provide commercial-off-the-shelf (COTS) products that meet its client’s needs without extensive customizations. The less customization that is required, the easier it will be to implement, validate, and update one’s LIMS. This improves the ability to reap the benefits of future product enhancements, giving one the best likelihood of future-proofing one’s investment.

*Mr. Leitham is Director, Product Development—Informatics, **Thermo Electron Corp.**, 1601 Cherry St., Ste. 1200, Philadelphia, PA 19102, U.S.A.; tel.: 215-964-6020; fax: 215-964-6021; e-mail: marketing.informatics@thermo.com.*