

A GUIDE TO SUCCESS

Enterprise Asset Management (EAM) System Implementation

Table of Contents

2	EXECUTIVE SUMMARY
3	INTRODUCTION
4	SUCCESS CULTURE
5	NEW SOFTWARE ENGINEERING CREATES NEW IMPLEMENTATION CHALLENGES
7	RECOMMENDATIONS

EXECUTIVE SUMMARY

Now more than ever, enterprise software customers are demanding increased value from their vendors and implementers before agreeing to open their purse strings. Successful enterprise asset management (EAM) implementations link a company's mission, its assets and its resources into a system that balances the needs of every important stakeholder. EAM vendors often provide the basic transaction platform for achieving that goal, but don't always provide the full complement of technologies and domain expertise required in every instance. Financial management and IT often dominate decision-making, but engineering and service requesters have effective veto power over EAM success. The long-term effectiveness of EAM implementations relies on close alignment with company operations and industry practices as well as a determined focus on authoritative performance benchmarks for setting goals and gauging progress. The cultural qualities and innovative technologies needed for consistent EAM success is more typically found in specialized implementers with extensive industry and asset category experience rather than with the software vendors themselves. Unsuccessful EAM projects are not so much bad software trying to become good as they are sick implementations trying to get well.

INTRODUCTION

Software doesn't wear out, implementations do.

Analysts and journalists have reported for years on the problems with enterprise software. Whether it boils down to a manufacturer's failure to deliver stable software or the quality of its implementations, one thing is clear – that the success of any software implementation is directly linked to key decisions that enterprises make with the external companies they hire to configure, integrate and deploy their systems. It has often been reported that many enterprises today just don't feel they receive enough value from their software vendors.

While the clamor isn't entirely new, it seems to be reaching a crescendo. On January 2, 2004, *The Wall Street Journal Europe* devoted its famous column one to the conflict between software buyers and sellers. In short, buyers want more value for their software dollars and are demanding that sellers change the way they do business.¹ Vendors are responding, but not quickly enough for some; and customers are defecting, usually at considerable cost to their bottom line. (In some cases, the entire cost of the software, licenses and annual maintenance has to be written off.) Why does this happen? As the *Journal* points out, "software doesn't wear out."² The obvious place to look, then, is the management decisions that affect software implementation practices.

It's fashionable among analysts to place some of the blame on vendor hype, but that doesn't explain why many software implementations succeed in spite of it. Applications sometimes fall short in the eyes of their users because they only managed to enlarge the silos of enterprise information rather than eliminate them altogether. A troubling fact is that even those applications that work don't always deliver enough value to justify high, and rising, software maintenance and upgrade costs. Vendors haven't done themselves any favors, sometimes acting like cable TV franchises that hike prices once they've captured a community's business.

Others believe that technology, specifically so-called Web architecture, is the answer to our problems with enterprise software. Yet history suggests that new technology typically falls short of its own hype and also brings unexpected new costs. The move from terminal/server to client/server technology in the 1990s is a case in point, and there are many indications that the same fate will befall so-called Web architectures that are now emerging. Also, a day in any large IT department will convince most that some of the best software implementations in use today employ extremely outdated technology.

...the success of any software implementation is directly linked to key decisions that enterprises make with the external companies they hire to configure, integrate and deploy their systems.

The fact is that applications often don't do what they promised – at any cost. This bitter fact played a key role in the multi-year decline of enterprise software license revenues worldwide. And, even though enterprise software sales are again rising, as the recent *Journal* article points out, vendors have had to make significant concessions to customer demands.³ Enterprise asset management (EAM) software was particularly hard hit by this wariness, posting declines in licensing earlier than the rest of the enterprise software industry and continuing the decline through 2003, even when other categories returned to double-digit growth.

Regardless of the technology or market trends, some software implementations succeed while others fail. It appears that successful enterprise software implementations align business objectives with technology, and at the nexus of those two lay the software implementation resources and methods. That is where we will look for answers.

¹ K. Delaney and D Bank, "Exploiting New Clout, Clients Are Saying No to Software Suppliers," *The Wall Street Journal Europe*, Jan 2-4, 2004. ² *Ibid.* ³ *Ibid.*

SUCCESS CULTURE

What's missing?

Stakeholder balance and minority rights.

Business schools teach that successful management requires meeting the needs of every important stakeholder both inside and outside the enterprise: labor, capital and customers. The first step in setting up a success culture is correctly identifying the important stakeholders and determining how to meet the needs of each one. An enterprise asset management project has four key stakeholders: financial managers, service requesters, skilled trades (contracted or in-house) and IT staff. Whether a part of the process or not, each has effective veto power over a project's success as many anecdotes from the files of failed EAM implementations attest. [See *"Stakeholders have the last word."*]

How to meet those needs, especially when they conflict with those of other stakeholders, and how to measure the extent to which they are being met is poorly understood and rarely practiced. Instead, power-driven implementations are the norm – that is,

stakeholders compete for influence over the process and the more politically powerful ones usually win the argument. Minority rights and opinions count for little in such implementations. Successful EAM implementations balance stakeholder needs and protect the less powerful.

What's missing? Performance tracking.

How do you know when you've reached a goal? An athlete doesn't win Olympic gold by wishing for it; rather, he must find ways to develop specific skills and build endurance. He must adopt a training schedule and measure his progress over time, perhaps setting interim goals to see how well he's tracking toward his ultimate goal. He must utilize advice from sports medicine professionals to ensure that his training does not endanger his own health. In consultation with coaches, he must compare himself with other athletes in his chosen sport to determine the level of competition he will face.

As obvious as that approach may be, many EAM implementations fail to attempt even rudimentary performance measurement and industry

STAKEHOLDERS HAVE THE LAST WORD

Regardless of the number of stakeholders involved, each has an element of power when it comes to determining whether a project is successful. The following examples speak to the importance of involving all stakeholders.

The back-up that wasn't.

The engineering department of a world-renowned teaching hospital in the Boston area chose to go it alone by setting up its own dedicated LAN rather than submit to IT oversight and associated costs. Because of an operator error, the department lost its entire maintenance history just as a critical audit was approaching. No problem, the engineers thought, because they had backed up their records on a regular basis for years. Unfortunately, they had never validated the restore process, and when it came time to retrieve the lost data, they learned too late that the back-up process was faulty and the data irretrievable.

We wish you hadn't asked.

During a field trip to view the reference site of a leading EAM vendor, the evaluation committee of a household name in passenger automobiles heard first from senior financial managers. The reports were glowing. The committee then met with plant managers at the site and received equally strong reports. As the committee was about to leave satisfied with what it had found, one member, representing the skilled trades, asked for permission to speak with maintenance engineers on the plant floor. When asked how they liked the new maintenance system, the plant engineers responded that the system, while implemented at the insistence of financial and plant managers, was not used by

plant engineering because it interfered with good maintenance and manufacturing practices and to use it would actually be counterproductive to their mission.

Can't we all just get along?

A leading EAM vendor booked the largest single license in its history with an agency of the U.S. government to maintain a major component of the North American transportation infrastructure. Many years later the system (though paid for) remains unimplemented, the victim of competing and irreconcilable agency agendas. The multi-million dollar project recently was re-bid in hopes of finding a "better" software solution.

benchmarking. After an initial ROI calculation – usually performed to satisfy financial managers that the project is sound – EAM implementations rarely collect base data for comparison purposes or include a rigorous set of quantifiable objectives that are tracked and reported by the new system. Even more rare are performance benchmarks that are based on a recognized service or maintenance industry authority. Successful implementations define their objectives and establish performance standards up front and track them relentlessly.

What's missing? Stakeholder compliance.

An enterprise organization is not a single conscious being, but rather, a collection of people who are walking more or less the same path. Some inevitably wander, and it is the job of the organization's leaders to maintain focus on the mission. Implementing an EAM project without ongoing stakeholder leadership is like conducting a war without making certain that orders are being carried out – the proverbial “fog of war,” where the link between plan and action is lost in muddled communications and the lack of reliable feedback from the front.

When stakeholder agendas remain hidden and managers do not attend to compliance, software projects falter. Third-party implementers play a key role in monitoring compliance and alerting project leaders when problems arise. Successful software implementations enlist strong, consistent leadership at all levels, not only to communicate clear objectives, but to maintain ongoing accountability among all participants.

What's missing?

More domain expertise (less software).

As is done with other types of enterprise software, major EAM vendors focus their best talent on developing the basic transaction platform for their application. Their management's focus is on developing new technology. So vendors discontinue support for older versions in order to force customers to upgrade, the software corollary to the automobile industry's planned obsolescence.

Not surprisingly, technology that squeezes more performance out of existing EAM implementations, even years after initial rollout, is not a top priority among software vendors. In contrast, external implementation specialists invest their best talent in developing technology and domain expertise with an eye toward business performance measurement and achieving the lowest possible application support costs over time. Successful EAM implementations seek out such specialists and employ their implementation technology to move a basic transaction platform to the next level, making it a really worthwhile business tool over the long haul.

NEW SOFTWARE ENGINEERING CREATES NEW IMPLEMENTATION CHALLENGES

Advanced, yes, but usable?

In high tech, the tendency is to focus on new technology innovation. But the PDA was not so much new technology as it was a breakthrough in usability. Certainly, when compared to Apple, the “Wintel” PC wasn't a computing innovation so much as a standard that rapidly reduced the cost of desktop computing functionality. Market-defining innovation doesn't always, or even often, come from new technology. The “love-me-love-my-technology” attitude of the past has given way to a sober look at both the upside and the downside of new technology. Customers are demanding technology innovation that is easy to use and trouble-free, not just new.

EAM developers give pride of place to new technology platforms including browser-based UIs, multi-tiered client/server architectures, XML interfaces and expanded application servers, operating systems and database support. Outside of some IT cost savings, available only after a significant investment in system upgrades and in-house organization skills, mainstream companies find few compelling reasons to jettison existing applications. A surge of new technologies, and the wrenching internal changes they may require, are driving many in IT to consider outsourcing their application management altogether.

CASE STUDY: A SUCCESS BECAUSE THEY PLAYED BY THE RULES

The U.S. Army builds a single point of entry for facilities management requests.

When the director of the Army's Fort Eustis installation wanted to consolidate several systems used to log service requests, schedule maintenance and account for maintenance expenses, they chose packaged asset management software that met most, but by no means all, of their needs. To build out the remaining features, they had a choice of doing it themselves with the tools provided in their application or working with someone else, essentially a dilemma faced by every enterprise software project.

The Army chose an implementer with deep experience in government operations and accounting practices but, more importantly, a set of productivity products that allowed much greater flexibility in designing business processes specific to its needs at the base. The tools enabled swift screen re-configurations, required- and restricted-data entry, user-defined conditions, virtual calculated fields and even entire custom processes with a low technical investment in programming or database table modifications.

Using an external rules engine to define customer-specific requirements saved money up front and also enabled a much higher degree of change management down the road. Because a rules engine makes business processes explicit, future application support personnel don't have to guess what the original implementers intended the next time business processes are under review. And because the engine adheres to the underlying application's published API, the Army can easily migrate the rules each time it upgrades. That represents an enormous support cost advantage as conditions change, software improves and new practices are required to meet the demands of war and peace.

Legitimate technology solutions are few and far between. And market-defining innovation addresses issues much broader than just IT deployment costs. Enterprises also want usable interfaces, easier upgrades and rules-based application configuration tools.

Your practices, our practices or best practices?

The difference between true best practices and "our practices" is an authoritative standard or industry benchmark. While giving lip service to best practices, no EAM vendor does an adequate job of delivering them. Almost no vendor provides much more than third-party support for continuous improvement techniques such as business intelligence, quality management or balanced scorecards. And no leading EAM product has ever included a user-configurable, best-practices reference model or rules engine, much less multi-scenario planning capabilities.

Defining the right way to do things is critical to the aforementioned goal setting and performance tracking. The "mile-wide-inch-deep" aspect of enterprise software occurs because developers must spread their R&D investments over the widest possible population. It's impossible to be an expert in every flavor of asset maintenance management, from nuclear power plants to home appliances. Successful EAM implementations are characterized by specialized knowledge of best practices in specific kinds of operations and asset classes (e.g., facilities, production, fleets, consumer durables, public works, IT.). Successful implementations are enabled in part by third-party rules, performance and compliance tracking tools not provided at depth by the software vendors themselves.

Is new technology ready for prime time?

Enterprises need to look closely at what they gain *as well as what they may lose* by deploying the early implementations of the latest application architectures because incomplete software engineering standards, while designed to lower IT deployment costs over time, may introduce a new set of configuration, upgrade and usability difficulties. Software vendors have been known to rush immature technology to market to gain an edge.

Innovative implementation specialists have stepped in with technology to plug the holes in EAM platform technology as new multi-level application architectures have introduced significant compatibility, performance, load balancing and availability issues. [See sidebar, "Implementer overcomes Web-architecture shortcomings."] Successful enterprise software implementations employ innovative third-party tools to mitigate the risk of using relatively immature platform technologies introduced in recent years by EAM vendors.

RECOMMENDATIONS

Leverage other people's experience.

Seek domain experts who have been down the path you intend to tread many times before. Look for demonstrable success in the largest, most complex maintenance and repair organizations similar to your own. No one can be expert in every type of operation, and off-the-shelf performance standards that apply to every situation do not exist. You need a specialist with experience to set goals and implement the needed performance measurement.

Successful projects depend on entire systems, not products.

No single EAM product has everything needed to insure project success. This is particularly true of newer "Web-architected" products because of that technology's relative immaturity. Supplement the product that will serve as your EAM transaction platform with appropriate third-party technologies that fill the gaps in your system. In particular, obtain technology that supports continuous improvement efforts such as performance measurement, best-practices benchmarks and change management.

Link every stakeholder's requirements to the system and negotiate compliance.

Ensure that business objectives will be met by balancing all key stakeholder interests, not just the more powerful ones. Incorporate performance measurement directly into the system, not as a separate process or afterthought. Deliver information agreed to by each stakeholder in a consistent, measurable and easily interpreted way as an ongoing check on compliance. Seek additional technologies such as reference models, rules engines and independent benchmarks from third parties when the EAM product vendors fall short.

When upgrading, consider a full re-implementation.

Unless your organization is a rare breed, current EAM implementations lack many of the qualities outlined above. Given the costs of upgrading any substantial enterprise application, take the opportunity to re-assess your success to date and re-specify business objectives, performance measurement and user compliance, linking them tightly to your implementation. Not only may you save your company a lot of money in process improvement, but you will almost surely ease the justification process for future EAM enhancements and upgrades.

Build the next level of performance into the current EAM system.

Companies continue to be relentless at cutting costs. Plan on it by setting business objectives for asset management that anticipate further streamlining. Don't assume that another EAM upgrade budget will miraculously appear somewhere down the road. Instead, embed continuous improvement goals in your system and begin collecting progress data from day one. When financial managers ask for justification for continued budget support, you want to be ready with answers, not guesses.

IMPLEMENTER OVERCOMES WEB-ARCHITECTURE SHORTCOMINGS

Java and page server technology have been a boon to IT departments, but something of a bane as well. Easy access to the text-based script that defines how screens look allows IT personnel to re-configure without special tools or training. But there's a downside. Early implementations of page server technology did not separate script that defines what screens display from script for critical business logic because an authoritative programming standard did not yet exist. Whereas today the Java community observes a standard known as Model 2/Struts, early Java page server technology co-mingles display and business script, making configuration and upgrades labor intensive and much more difficult.

Total Resource Management, Inc. (TRM) of Alexandria, Virginia, is a leading specialist in facilities and public works asset management. TRM met the challenge of a newly minted EAM product by developing an intuitive screen configuration tool. While text-based script may be universally accessible, it is not particularly easy to use. TRM created a visual navigation layer so that displays can be reconfigured using more usable drag-and-drop techniques, even by the end users themselves.

More importantly, the tool overcomes the danger of potentially disastrous modifications to application business logic by insulating logic from changes to the screens. IT gets the security it demands, and end users get an extremely flexible system that can be easily configured to meet their objectives without undue deployment, support and upgrade expense. Specialized implementers like TRM bring both domain knowledge and critical technology that combine to reduce the risk of EAM projects.

Council for Service/Asset Management

C-SAM (www.c-sam.org) research and education aim to develop one set of standards to describe and manage (non-financial) enterprise assets as a single domain rather than separate classes that today enjoy few common conventions and processes ... a shared language to underlie the measurement of all asset performance.

Asset owners and suppliers alike benefit because existing standards efforts, though piecemeal, have already proven a boon to both enterprise asset optimization and supplier responsiveness. The burgeoning power and reach of technology permit an unprecedented level of information sharing, paving the way for collaboration and spreading the effort required. The council's mission is to enlist participation from all stakeholders, to pool the best business practices and to achieve widespread endorsement for agreed-to standards.



TRM Headquarters

510 King Street
Suite 300
Alexandria, VA 22314
Phone: 877-548-5100
Fax: 703-548-6211

Advanced Technology Centers

Chesapeake, Virginia
Honolulu, Hawaii
San Diego, California
Seattle, Washington