



# Managing EH&S Software Project Risks

Organizations today increasingly face environment, health, and safety (EH&S) challenges. To help meet these challenges, many enterprises are seeking software solutions that will help them remain competitive and avoid using additional staff resources. But what should organizations know before taking on significant in-house information technology (IT) projects? Read on to learn a few ways to manage EH&S software project risks.

## STRATEGIC ALIGNMENT

In large and small organizations alike, it is not uncommon for IT staff to begin a software initiative only to discover that it has little chance for success. Your organization's core business strongly influences the success of any initiative. If the software project aligns with the overall business strategy, then it has a good chance for success. Similarly, if the organization views IT as a strategic function—much more than a staff or support function—then the project has a good chance of success (see August 2006 Survey, *CIO Insight*, www.cioinsight.com).

## KNOW YOUR CAPABILITIES

### Development Framework and Mindset

In-house software development (as opposed to purchasing commercial software) requires focus, structure, and the discipline to follow rigorous methodologies. Likewise, in-house development may require retooling IT governance, adding staff with the appropriate skill sets, and establishing accepted IT management practices and systems.

EH&S software development is not for everyone. The software life cycle spans the time from concept through development, testing, rollout, and support. The cycle then starts again as users demand enhancements and new features to keep up with changing EH&S needs. It's a continuous process.

## Multiple Skills

It is important to note that end-user needs drive software development; the IT development organization should not drive end-user needs. While internal IT departments may have talented business analysts and programmers who understand business systems, they must also fully understand EH&S needs. Further, they must be able to translate EH&S needs into clear and concise software requirements that resonate with EH&S staff, operations, and other stakeholders. The project team not only needs to be multidisciplinary, but also fluent in several technical "languages."

## Change Management

Since EH&S software initiatives span months or even years, change is bound to happen with people, processes, and technology. Over the lifetime of a project, people move on. Be sure that your team includes redundant expertise, so that the project does not rely too heavily on one or two individuals.

Regulatory-driven processes, especially EH&S, continually change. Be sure to put processes in place to keep regulatory-driven elements of your EH&S software up to date and reliable. If your software application incorporates regulatory citations, seek a regulatory reference service that integrates with your software and provides frequent updates. This also helps you identify any changes that need to be made to your EH&S business processes.

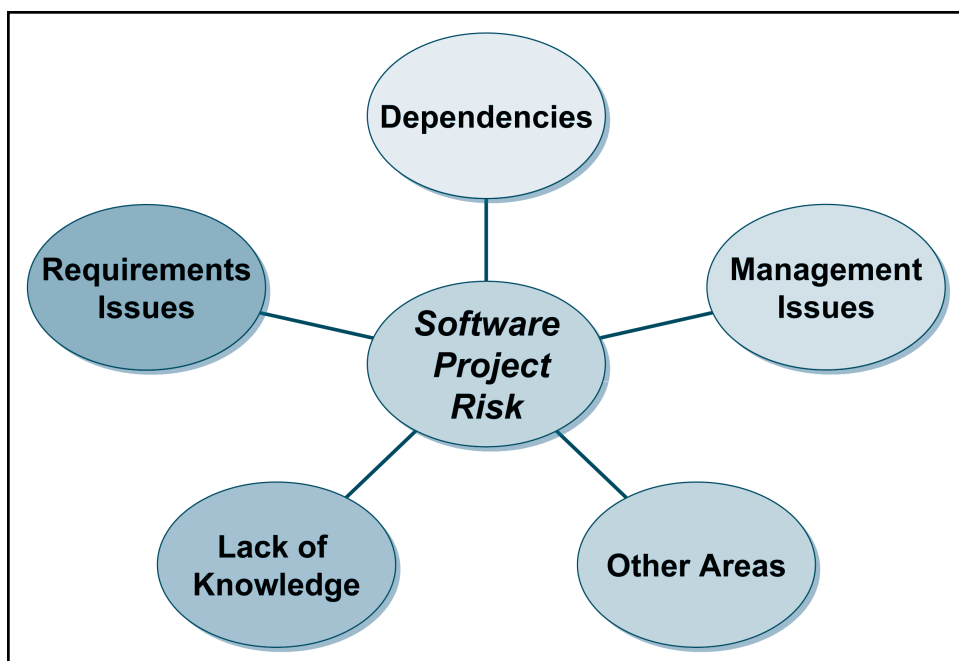
**End-user needs drive software development; the IT development organization should not drive end-user needs.**

Technology continues to change as well. Consider the useful life of your hardware and software and develop strategies for upgrades and for system end-of-life. If you plan to build a system, invest adequate resources to handle both the initial build and keeping the application current when technology changes.

## RISKY BUSINESS

### What Is Risk?

Risk is inherent in any software effort, whether you build it in-house or purchase an off-the-shelf commercial software application. Simply defined, a risk is a problem that can cause a loss or threaten a project's success. Risk may result from a defect, a failure, or any situation or event that increases the



**Figure 1.** Types of software project risks. *Source:* Wiegars, K. Know Your Enemy: Software Risk Management, *Software Development*, October 1998, www.processimpact.com.

and external subcontractor relationships; inter-component or inter-group dependencies; the availability of trained, experienced people; and reuse from one project to the next.

**Requirements Issues.** The project team must clarify requirements early to avoid building the right product poorly, or building the wrong product. Requirements risk factors include a lack of clear product vision; a lack of agreement on product requirements, or unprioritized requirements; a new market with uncertain needs; new applications with uncertain requirements, or rapidly changing requirements; ineffective requirements change management process; and inadequate impact analysis of requirements changes.

cost of the system, delays its delivery, or reduces its benefits.

Risks can adversely affect the project cost, schedule, technical success, software quality, or project team morale. A critical risk can jeopardize the successful delivery and operation of the system (see [www.mosaicinc.com/defectmanagement/risk.htm](http://www.mosaicinc.com/defectmanagement/risk.htm) for an analysis of risk). To ensure success, you must accept that risk is present, and that managing it is crucial. This might require a cultural shift in your organization.

### Risk Management

Managing risk helps reduce uncertainty. Risk management is the process of identifying, addressing, and eliminating potential problems before they can damage a project. A typical risk management process includes identifying risks, assigning each risk a probability, prioritizing risks, and developing controls to manage risks. One approach places software development risks into five categories: dependencies, requirement issues, management issues, lack of knowledge, and other risk areas (see Figure 1).

**Dependencies.** Many project risks stem from reliance on outside agencies or on factors that we cannot control. These include customer-furnished items or information; internal

**Management Issues.** Management issues can derail a project. Use defined project management processes and spell out roles and responsibilities to mitigate the following risks: inadequate planning and task identification; unclear project ownership and decision-making; unrealistic commitments (sometimes made for the wrong reasons); managers or customers with unrealistic expectations; staff personality conflicts; and poor communication.

**Lack of Knowledge.** The project team may not have the skills required to execute the project, especially with technology changing so quickly. Avoid the following vulnerabilities: inadequate training; poor understanding of methods, tools, and techniques; inadequate application domain experience; new, untested technologies or development methods; and ineffective, poorly documented, or neglected processes.

**Other Risk Areas.** Other risk areas to consider include the availability of development or testing equipment and facilities; the ability to acquire resources with critical skills; turnover of essential personnel; unachievable performance requirements; problems with language translations and product internationalization; and technical approaches that may not work.

Many organizations are choosing to undertake software development initiatives, whether in-house or by teaming with commercial software vendors and implementation firms, to help meet growing EH&S challenges. If you are considering this route, be sure to check for project alignment with your organization's business strategy, know your capabilities, and identify the risks. Focus on what your organization does well, seek help to fill the gaps, and forge ahead with your eyes wide open. **em**



Jill Barson Gilbert, QEP, is president of Lexicon Systems, LLC. She helps organizations increase business value by designing and implementing EH&S management solutions that leverage technology. E-mail: [JBGilbert@Lexicon-Systems.com](mailto:JBGilbert@Lexicon-Systems.com).