Internal Audit’s Role in ERP Implementations

Warren P. Chan, CISA, and Arlene M. Manahan, CISA

Implementing a new enterprise resource planning (ERP) system is inherently risky for any organization. With more than 90 percent of implementation projects completed late or over budget, or both, it is crucial for organizations to recognize the risks and define appropriate implementation strategies. Internal audit’s unique role in an organization makes it an ideal resource for any implementation project. In this article, we describe the various phases of an ERP implementation and identify the areas where internal audit should have an active role.

Before embarking on an ERP implementation project, an organization must balance the benefits of a new ERP system with the costs of implementing and maintaining it. With a well-designed project plan, the implementation cost can be estimated, but that assumes that there will be no major project delays or resource skill issues. Projects that are mismanaged or that don’t include sufficient skilled resources can easily incur costs that exceed the project’s anticipated return on investment. Following long, painful, and expensive implementations, some companies have difficulty identifying measurable benefits. One mismanaged ERP implementation left a southeastern electronics manufacturer unable to accept deliveries and nearly closed a plant.

Often there is a great difference between the optimistic expectations for an ERP system and the actual value derived from it after implementation. A study cited in a Panorama Consulting Group 2008 ERP report noted that 93 percent of ERP projects fail to go live on time and on budget. In addition, The Standish Group, in its “CHAOS Summary” report for 2009, estimated that only 32 percent of IT projects are completed successfully. According to the report, 44 percent of projects are completed late, over budget, or with less than the required features and functions. Furthermore, 24 percent of projects are canceled prior to completion or are delivered but never used. It is, therefore, critical for an organization to be as certain as possible that project risks are well understood and that it deploys the appropriate strategies, tactics, and controls to minimize the chances of the risks becoming reality.
Internal Audit’s Role

The internal audit function can help identify, review, and provide recommendations for key controls associated with the project and can provide assurance that the ERP system will support business processes and enforce business controls on an ongoing basis. The use of collaborative internal auditors on all critical phases of an ERP project is the best approach to increasing the likelihood of a successful ERP deployment.

Throughout an ERP implementation, internal audit can help identify and communicate risks by having them addressed throughout the project instead of as an afterthought. By understanding the major phases and objectives of an ERP implementation, internal audit can objectively raise issues that, if overlooked, could jeopardize a project’s success. Internal auditors can also articulate the risks from a management perspective.

Success for ERP implementations, like all projects, is not determined solely by whether the project was completed on time and on budget. The quality of the final product and its alignment with management’s desired objectives have long-lasting impact beyond the initial project costs. Fixing errors after the system goes live is more costly than correctly implementing the system in the first place. Throughout the implementation, internal audit should have a vital role in verifying that project controls and best practices are followed. This role greatly reduces the risk of failure resulting from poorly defined methodologies or weak enforcement of project controls.

An ERP system is usually the system of record for most of an organization’s accounting function. It is critical that the system enforces controls for every relevant business process. For this reason, the organization needs to focus on meeting project management (PM) and software development life cycle (SDLC) controls requirements while also considering appropriate business process controls for the design and operation of the ERP system.
Phases of Implementation

PM/SDLC controls and business process controls need to be considered throughout each phase of the implementation, from inception to post-go-live support. While implementation methodologies may differ in the names of phases or may group certain phases together, all major implementation methodologies follow a similar sequence of phases:

- Strategy
- Planning
- Analysis and Design
- Development
- Testing
- Training
- Deployment
- Post-Go-Live

Strategy

During the strategy phase, key executives make decisions including what software to implement, whether to perform the work in-house or to outsource it, and the approximate timeline for the project.

The internal audit function can assist in the decision-making process by assessing whether the selected software is capable of enforcing controls, and if the group performing the implementation has a sound methodology that incorporates a control orientation with an emphasis on meeting business requirements. A strong implementation methodology mitigates overall project risks by carefully defining roles, tasks, and goals.

Planning

In this phase, the project team, with input from the company’s management and IT management, creates a detailed project plan, including budgets and timelines. A commonly overlooked step during this process is verifying that minimum standards for project deliverables are defined. Unfortunately, it is common for a project plan to call for a deliverable such as “business requirements” without providing any guidelines. Failing to set a minimum standard puts at risk all deliverables that are dependent on the initial or prerequisite deliverable.

Planning should include internal audit’s evaluation of the intended deliverables and a review to confirm that the appropriate approvals have been obtained.

For example, if the business requirements deliverable fails to define all the pertinent business expectations of the system, the functional and technical specifications likely will fail to address the business needs of the organization. More than likely, this failure will result in the configuration or custom programming not meeting the actual requirements. This, in turn, can cause significant amounts of time and money to be lost in trying to correct problems discovered toward the end of the development cycle or, worse, after implementation.
While setting standards is a prerequisite, it is equally important to enforce the standards. This can be accomplished easily with a review and sign-off process. Again, it is necessary to set a standard for how reviews are conducted to establish that designated reviewers are not just rubber-stamping deliverables that require their approval. Review and approval standards must include a reviewer who is proficient in the subject matter that the deliverable covers, a process for comparing budget to actual, and a process for verifying that the deliverable meets not only project needs but also the overall needs of the organization.

Common deliverables for which minimum standards should be defined include:
- Business requirements
- Functional specifications
- Technical specifications
- Configuration documents
- Test plans and scripts
- Training plans and training material
- Programming documentation (code comments, for example)
- Go-live checklist

Internal auditors can assist by verifying that a reasonable minimum standard is defined for each major deliverable and that the review and approval process is robust enough to provide quality deliverables. During the later phases, after the work is completed, internal auditors can perform a review to determine whether these controls are being appropriately enforced.

**Analysis and Design**

During the analysis and design phase, the project team identifies the business requirements, evaluates gaps in current business processes and application functionality, determines solutions for those gaps (by changing business processes or customizing the applications), determines the correct configuration settings for the application, and documents the functional specifications of any required customization.

Traditionally, internal audit’s role in this phase of the project has been limited. However, this can be the most important phase to have internal audit involved. As the current business processes are compared with the future ERP system's capabilities, internal audit can verify that:

- All key controls can be implemented in the ERP system;
- Any automated controls the ERP system can provide are used to strengthen the control environment;
- Controls that cannot be enforced by the ERP system are identified, and appropriate manual compensating controls are designed;
- Existing controls are re-evaluated and redesigned for any modified business process; and
- Application access processes and controls, including user provisioning and segregation of duties, are developed and documented.
Development
During this phase, the implementation team configures the application and develops any customizations.

Internal audit has a role in verifying that project management and SDLC controls are performed and enforced adequately by the team’s project managers.

Testing
During testing, the implementation team performs detailed tests to determine if the ERP system is functioning as designed and is meeting all of the documented business and control requirements.

In addition to playing its ongoing role of evaluating the enforcement of project management controls, internal audit can assist with the testing to confirm that the ERP system is enforcing controls effectively and that any necessary evidence is being captured for audits or compliance requirements (Sarbanes-Oxley Act Section 404 requirements, for example).

Internal audit should not limit its review to automated controls. At this point, the project team, working with the business process owners, should have designed and documented manual controls associated with the ERP system. These controls include manual business functions, such as reviewing and approving reports, as well as support functions, like requesting and approving user access. Internal audit should determine whether these controls are being enforced during the testing phase. It is very common for testing to occur without validating the access enforcement security model that will be used in production. Failure to test the application with the appropriate security enforcement controls may lead to incorrect conclusions regarding security and proper segregation of system function capabilities.

Also, experience gained in understanding and collecting evidence from the new ERP system can increase efficiencies for future audits of the system.
Training
Training is not usually a distinct phase during the implementation. Generally, organizations conduct training simultaneously with other phases, starting as early as the project kickoff and as late as the testing phase. During training, end users learn about the correct operation of the new application and IT support personnel acquire the requisite skills for ongoing support and maintenance of the ERP system.

Even if the software is configured perfectly and all customizations are implemented properly, a project can fail after it goes live if the end users or the IT support personnel have not been trained adequately. For this reason, internal audit should verify what training was conducted as well as evaluating the quality and reasonableness of the training.

Deployment
During the deployment phase, the implementation team performs the cutover from the old application to the newly implemented solution.

To determine that the decision to go live with the new ERP system is appropriate, internal audit should:

■ Verify that all outstanding project issues are adequately addressed;
■ Confirm that a detailed go-live checklist has been created and approved by project management and stakeholders, and that no “show-stopper” issues have been encountered prior to implementation; and
■ Verify that a detailed contingency plan exists.

Post-Go-Live
After the new ERP system goes live, the implementation team provides support for any issues encountered and addresses any noncritical issues that were identified before deployment.

Internal audit should perform a post-implementation review to verify that the controls are operating effectively. Internal audit should test business process controls enforced by the ERP application as well as the related application support process controls. In general, the audit should be scheduled approximately three months after the system goes live to allow sufficient time for the system to complete daily, monthly, and quarterly transactions prior to the control testing.
Three Internal Audit Approaches

In practice, there are three common approaches to internal audit’s involvement in ERP implementations:

1. Internal audit is involved in each phase of implementation;
2. Internal audit is involved during or after testing is completed, and before going live; and
3. Internal audit is involved only after the ERP system has gone live.

Clearly the first approach, as we have described, is the most effective, least risky, and least costly.
Implementation Success

A majority of ERP implementations are not completed successfully. Involving internal audit in every phase throughout the project will save significant time and money as well as reduce the risks inherent in any such project, thus increasing the implementation’s chance for success.


4 “CHAOS Summary 2009,” The Standish Group, April 23, 2009


7 Robin Basham, “Procedure Guidelines and Controls Documentation: SDLC Controls in COBIT 4.0”


Contact Information

Warren Chan is a principal with Crowe Horwath LLP in the Chicago office. He can be reached at 630.586.5135 or warren.chan@crowehorwath.com.

Arlene Manahan is with Crowe Horwath LLP in the Chicago office. She can be reached at 630.575.4332 or arlene.manahan@crowehorwath.com.