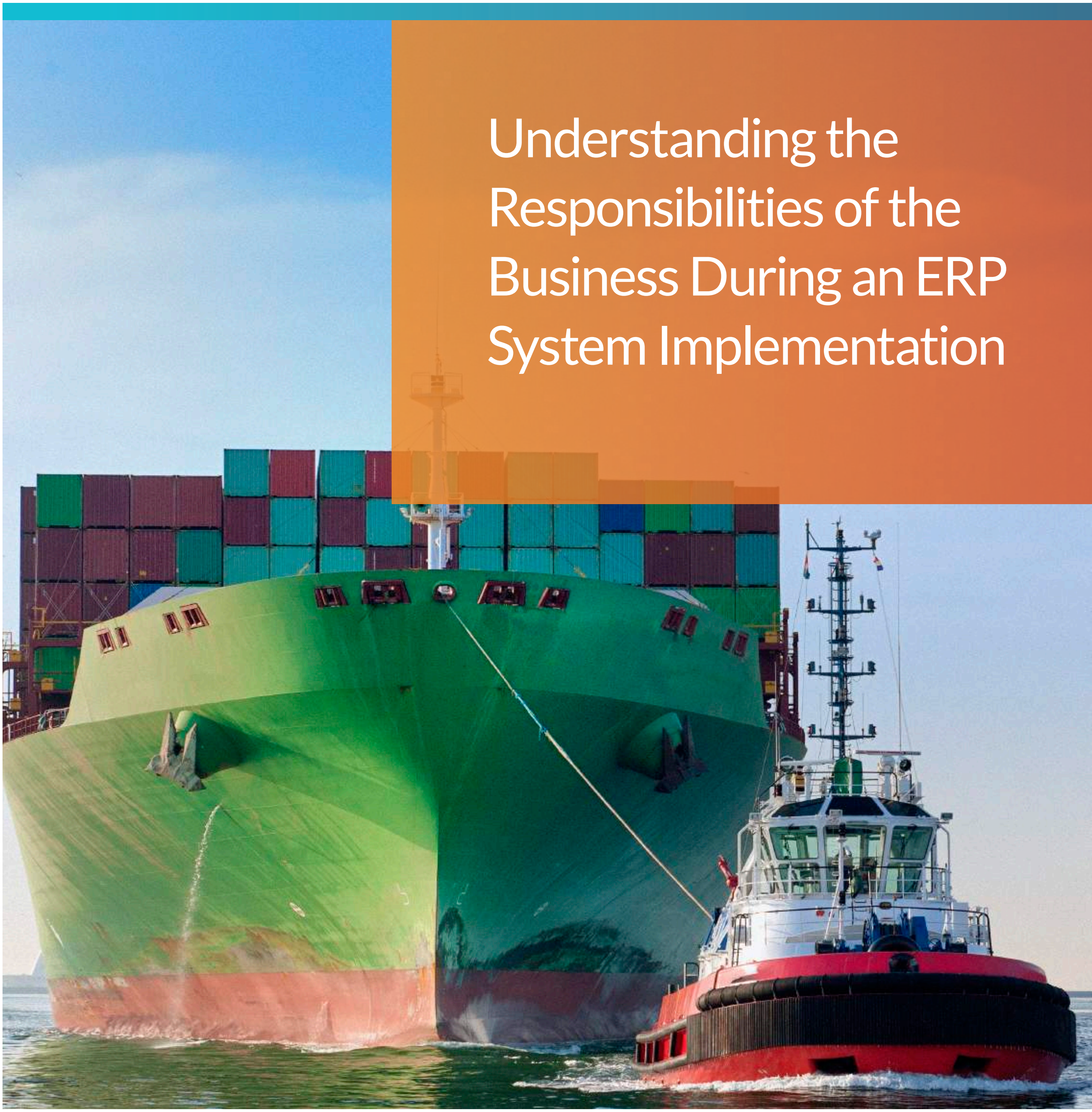
protiviti®

Face the Future with Confidence

Understanding the Responsibilities of the Business During an ERP System Implementation



Internal Audit, Risk, Business & Technology Consulting

Introduction

As organisations implement new enterprise resource planning (ERP) systems to automate processes, it is becoming increasingly critical not just for IT, but also for the business units themselves, to understand their central role in the overall success of these initiatives. The implementation of an ERP system or any other major IT system should never be viewed as just an IT project because, ultimately, it is a business project with business objectives.

ERP systems must be designed from the outset to support the business. Organisations cannot expect system integrators to develop these designs alone. Integrators are technical experts – not business process experts. This is why the business should be responsible for, among other things, defining the vision and operational expectations for the future state of each business process that the new system will impact.

If the business fails to fully embrace its role in a system implementation, it can potentially face significant and costly risks. Interminable project delays and budget overruns, business disruption post go-live and insufficient user adoption are just a few common examples. Even when the project is supported by a strong system integrator, it is critical for the business to assume responsibility for key activities before, during and after the implementation. The seven areas below are the responsibility of the business. We discuss each of them in the sections that follow.

01 Program Management and Governance

02 Business Process Readiness and Solution Design

03 Organisational Change Enablement

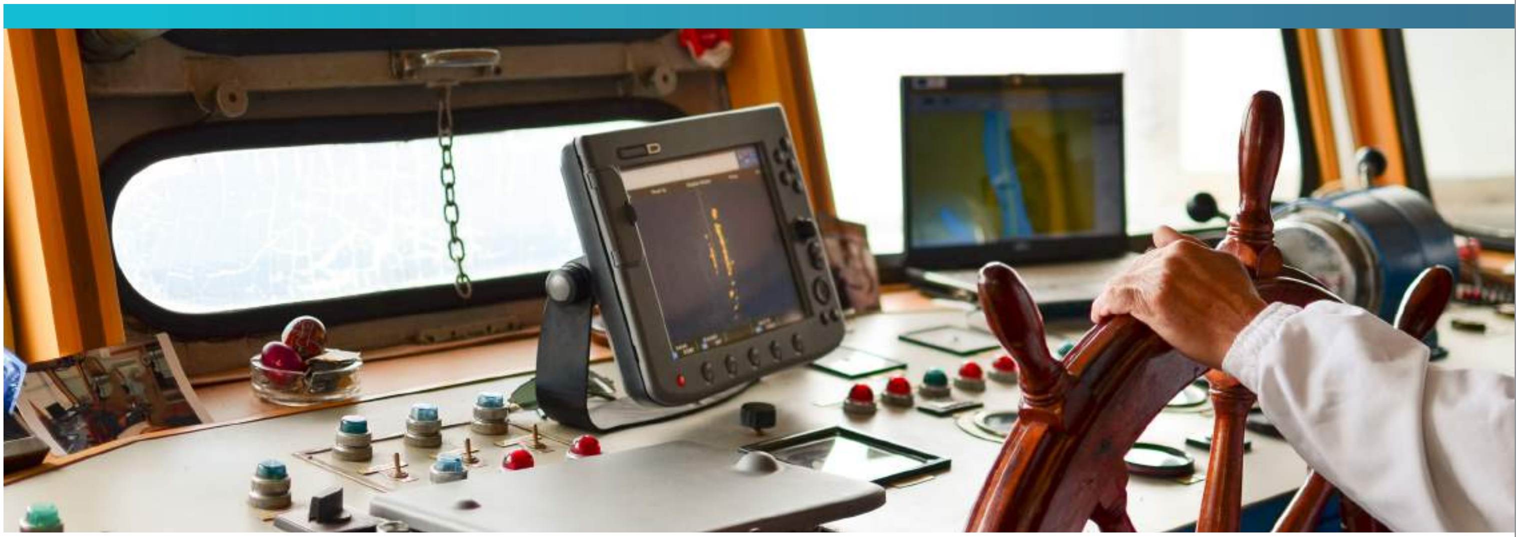
04 User Acceptance Testing (UAT)

05 Data Conversion

06 Data Governance

07 Business Intelligence and Reporting

This paper outlines the steps that a business can and should take in each of these seven areas. These actions will help to ensure that, once implemented, the new system will achieve the level of process improvement and automation, data quality, reporting and user enablement required to meet the needs and expectations of the business process owners. We anticipate ERP business sponsors will be able to use this document in designing and planning their ERP implementation programs and, at minimum, leverage it as a checklist for taking the essential steps to make their ERP program a success. It is an opportunity to learn from others, who themselves have learned these lessons the hard way.



Program Management and Governance

At a Glance

Key tasks for program management and governance include the following:

- Establish project management office (PMO) structure, roles and responsibilities.
- Define the overall program governance.
- Validate the completeness of the program plan.
- Expand the governance structure to enable a clear path for real-time decision-making.
- Establish continual monitoring and communication of critical program risks.
- Define a "single source of truth" for program status communication.

Most system integration firms will provide project management capabilities to oversee their teams' activities, but many will stop short of providing the level of management needed to oversee the end-to-end implementation.

Common gaps in a system integrator's management structure include oversight of internal business and IT resources, management of other vendors, and open engagement with company leadership (e.g., on risks and issues).

Achieving the proper level of implementation oversight requires a more robust program management approach. Companies should also institute a comprehensive governance structure for the entire program.

Key program management and governance tasks for the business include:

1. **Establishing the structure, roles and responsibilities for the PMO, starting with designation of a competent and experienced program manager to oversee the implementation.** A critical step in this is defining the reporting structures and responsibilities expected of vendor-supplied project managers. In addition, the program manager likely will require capable support personnel to assist with vital PMO tasks such as risk management and change control. The PMO needs to report to the right sponsor or group who owns the project, is committed to its success and has enough clout and authority within the organisation.
2. **Defining the overall program governance structure, beginning with the critical roles and responsibilities expected of business, IT and vendor personnel involved in the implementation.** Critical governance functions, such as a steering committee, also should be established and initiated. This process includes establishing a regular cadence for meetings and defining areas for oversight and monitoring. The meetings should have targeted agendas to help maximise their value and ensure they are more than just status updates.
3. **Validating completeness of the program plan to ensure it appropriately represents all parties and the true critical path.** A system integrator-centric plan often fails to consider the true impact and effort required for critical business tasks like UAT and data validation. The PMO should also maintain and continually evaluate the program plan throughout the implementation.

4. **Expanding the governance structure (defined in task 2) to enable a clear path for real-time decision-making.** Even though this capability is less critical in the early stages of the implementation, it becomes essential as the program nears completion. Rapid escalation resolution also may be required to preserve the schedule.

5. **Establishing continuous monitoring and communication of critical program risks.** This process should include establishing a risk score card as well as a methodology for regular/continual assessment of risk factors and risk mitigation efforts. Risk management also must be embedded across all functional areas within the program (e.g., UAT, data validation).

6. **Defining a "single source of truth" for program status communication.** This single source of truth should integrate perspectives from business, IT and vendor teams to provide a true picture to management of project health.

The bottom line: A typical ERP implementation is complex and cross-functional. It therefore requires a comprehensive program plan, clear assignment of responsibilities to competent and accountable work stream leads, and an effective, real-time governance process that facilitates tracking of commitments and identification and resolution of program risks and issues.

Business Process Readiness and Solution Design

At a Glance

To meet operational expectations for the new system, the business should design process models for the end-to-end future state of each business process that the new system will impact. This overarching business process vision is called the "solution design."

The process models and underlying solution designs from the business will help system integrators focus on system blueprinting rather than designing future processes, which typically is not their core area of expertise.

The system integrator is usually a technical expert and not a business process expert. Therefore, the business must be responsible for defining the vision and operational expectations of the new system with regard to each business process. Specifically, the business must ensure that the technical solution the system integrator proposes will satisfy the business process vision and future-state goals.

This requires the business to design process models for the end-to-end future state – ideally, prior to the system integrator's arrival on-site. This is what we call business process readiness.

The process models represent the conceptual representation of processes and should identify:

- Which activities will continue to be purely manual
- Which activities will require human interaction with the future system
- Which activities will be performed automatically by the system
- Who will execute the activities that are not performed automatically by the system

Any business rules (e.g., customer price calculation, cost allocation, tax computation) governing the above activities must be detailed. If the future system will be integrated with other existing systems, the data attribute requirements and event triggers for these integrations should be documented. The business should also outline any contingencies or exception workflows, and determine which people in the organisation will be responsible for taking action when something goes wrong.

In short, a design process model provides the solution to carry out future process activities and defines the related organisational structure. This overarching business process vision is the "solution design." At the outset of the implementation, the process models and underlying solution designs will help the system integrator to focus on system blueprinting rather than trying to design future processes, which is not their core expertise. While the solution design provides the components of the future system architecture, business process readiness presents the process models, including the activities and their organisation along with data flows and controls.

During the implementation phase, the solution design should serve as a reference for evaluating the technical designs produced by the system integrator during the early stages of the implementation. In addition:

- Critical gaps must be addressed immediately.
- Open questions from process owners must be tracked and resolved throughout the course of the implementation. This should be done in partnership with the system integrator's solution architect.
- Manual workarounds must be thoroughly evaluated to ensure future processes do not end up broken, incompletely automated or otherwise suboptimal.

The solution design should also serve to guide testing plans, and applicable documentation can be leveraged for training purposes. Preparing the solution design up front helps to ensure that 1) the system integrator is building the system according to management's vision, 2) the system is properly validated and will support the company as envisioned, and 3) users will be familiar with the solution and ready to use it effectively upon go-live.

Potential Mid-Implementation Challenges

Many companies expect the system integrator to lead business process readiness and solution design as part of the blueprinting phase during implementation. Often, business process owners begin to raise concerns midway through the implementation life cycle because they don't see a detailed future business process, or they perceive a disconnect between the solution and required business process activities, or they see gaps in the end-to-end design, or they simply don't understand the solution yet due to its technical complexity.

How does the business mitigate the inherent risks of this situation in which business owners justifiably lack confidence that the system will meet their needs? In these situations, we often recommend that the business take time to create the business process models and related solution design, even though the implementation is at an advanced stage. This is an essential exercise, as it will help to ensure that business process owners have a reliable and understandable reference to review as well as an opportunity to provide feedback on and approve the system integrator's recommended technical design. The exercise also provides a sound basis for subsequent user acceptance testing. This corrective action may meet significant resistance given the perceived delay in the project. It requires strong sponsorship by executive management, buy-in from the process owners and consensus from the system integrator.

Organisational Change Enablement

At a Glance

System integrators typically train only a key subset of users toward the end of a system implementation. Therefore, the business must plan for and address other critical activities, such as developing policies and procedures and defining the roles and responsibilities of individuals across the organisation. Additionally, the business must deliver practical training to the remaining user community, using a comprehensive enablement plan that:

- Addresses critical process adoption
- Defines metrics to measure adoption
- Outlines a communication strategy
- Periodically assesses stakeholders' concerns and commitment

The business must also be prepared to monitor user adoption and process efficiency once the new system goes live.

As the solution design is established, the organisational impact of system and process changes must be determined, assessed and planned for to ensure that the anticipated benefits are realised. As a first step, the changes must be inventoried, and the impact to the organisation must be assessed. It is also important to perform an analysis of stakeholders and their needs and to develop change enablement strategies and activities to ensure changes are implemented effectively. Ultimately, the goal is the development of a change enablement plan that will raise awareness with key stakeholders, obtain their buy-in, and ensure their commitment to support the changes and the performance improvement objectives of the initiative.

Typically, a system integrator's vision of organisational change enablement is limited to "system training." Training alone, however, is not sufficient, as it does not address the required activities that must be performed to support critical changes, including those related to the development of policies and procedures and definition of the roles and responsibilities of individuals across the organisation.

Early in the implementation planning stages, the business should review the system integrator's plans for supporting the user community. This should include activities planned during the implementation (e.g., communications and training) and activities subsequent to go-live, such as go-live support. Support can be provided through documentation (e.g., policies, procedures, job aids) or specialised tools.

Even though system integrators usually will train a key subset of users toward the end of an implementation, the business must still plan for and deliver training to the remaining user community. This training must fall within a comprehensive change enablement plan that:

- Analyses the impact on stakeholders early in the project life cycle
- Develops strategies to address barriers and ensure effective adoption
- Addresses how the critical process changes will be adopted by the organisation
- Defines metrics for measuring adoption and its success
- Includes a communication and training strategy
- Periodically assesses stakeholders' concerns and commitment to changes
- Addresses post-go-live support, not necessarily in the traditional sense but to see what is working (effective adoption), and identifies remaining optimisation opportunities

Note: Post go-live, the business is solely responsible for monitoring user adoption and the efficiency of the resulting processes. Any process breakdowns must be remediated only after a thorough root-cause analysis.

User Acceptance Testing (UAT)

At a Glance

UAT is the final and most important phase of system testing, and it is designed to ensure the new system will support the business. It begins with a comprehensive solution design that serves as the blueprint for testing.

A specific population of real data points must be incorporated into test cases and made available for test execution.

The business must assume responsibility for UAT and follow a number of critical steps in the UAT preparation and execution phases.

The business also must be engaged in testing cycles prior to UAT, even though the system integrator is primarily responsible for those testing phases.

The purpose of UAT is to test business processes end-to-end after system configuration is complete. A focused UAT phase that goes beyond prior functional and technical testing phases helps ensure the implemented system design can support the business effectively post go-live.

UAT differs substantially from prior phases of system testing, such as the conference room pilot (CRP) or system integration test (SIT) phases, which are more focused on configuration validation and development of specific functional and technical requirements. These earlier testing phases do not typically provide the user base with a complete view of the end-to-end, future-state processes that the new system supports.

Effective UAT starts with a comprehensive solution design that serves as a blueprint for testing. The solution design should form the basis for defining the test scenarios, cases and data variations used within the UAT phase. Without a design-centric approach, UAT often devolves into a fragmented repetition of CRP or SIT testing that provides some end-to-end validation, but ultimately falls short of the validation business process owners require. Similarly, UAT progression should be measured and reported in business terms (i.e., orders and phase gates vs. functional points).

Without question, UAT is vitally important. The business must assume responsibility for UAT planning and execution. To do so, it should follow a number of critical steps throughout the UAT preparation and execution phases outlined below.

UAT Preparation

1. **Create a UAT management team, led by an experienced project manager, to oversee planning and execution.** The team should include members whose roles are focused on metrics and defects.
2. **Draft test scenarios, based on the solution design, that cover all business processes end-to-end.** Be sure to validate their comprehensiveness with process owners. The population of test scenarios should include a subset that validates the strategic objectives of the project (e.g., specified improvements in process or system performance). A specific population of real data points (e.g., purchase orders, customer orders) should accompany each scenario and be tested end-to-end. Ideally, the data points that have been processed in the existing systems should be used as data points to support the test cases. This approach makes it possible to use the data in the existing systems as a baseline to assess test results. To ensure it is comprehensive, the set of test scenarios and test data should also include process anomalies and exceptions. Do not fall into the trap of testing the straightforward, “happy path” cases only.

3. **Plan the UAT execution schedule, identifying individual testers and the sequence of scenarios, to ensure efficient use of the testers' time.** UAT planning should accommodate potential inefficiencies relative to prior phases of testing because it requires cross-functional participation in near-real time to simulate the future-state process effectively.

4. **Identify dependencies on other tracks of the implementation (see further information below on data conversion and system testing) and the actions required to ensure efficient coordination.** Availability of interfaced and converted data for UAT, and identification of that data (as well as other prepared data), are critical path items for effective UAT.

5. **Define the UAT entrance and exit criteria and facilitate sign-off by implementation leads and overall program leadership.** The criteria should be quantified where possible; however, the team should acknowledge that the criteria serve more as guidance than hard-and-fast rules for go/no-go decisions.

6. **Prepare the UAT execution materials that testers will use, incorporating specific system instructions from the system integrator.** In some cases, UAT may leverage the same scripts that were used in prior testing phases. However, these scripts should be reoriented/reorganised to better align with the UAT testing execution approach (i.e., scenarios, cases and data).

7. **Facilitate logistical preparations, including facilities and travel for out-of-state or international testers.** It is critical to collocate testers involved in common processes.

8. **Define the reporting metrics required to track progress during UAT execution.**

UAT Execution

1. **Manage the day-to-day execution of UAT.** This includes daily replanning and reprioritisation and holding people accountable for timely execution of testing and defect resolution.

2. **Facilitate UAT entrance, tester kickoff and UAT exit meetings.**

3. **Publish a daily update of the UAT status based on predefined, business-oriented metrics.** Timely tracking of results is a critical input to the metrics. It is also essential to the day-to-day governance, corrective actions and replanning required to keep UAT on track and use testing resources effectively. The business can facilitate results tracking by using a testing management tool; if one does not exist, manual effort will be substantial.

4. **Monitor emerging risks and collaborate with the PMO and relevant stakeholders to determine how best to mitigate them.**

5. **Coordinate, on a daily basis, defect management between the system integrator, the testers and the IT department.**

6. **Lead daily defect prioritisation and defect status meetings with process owners and other stakeholders.** Regeneration of test data is a potential concern for retest; the defect team should pay attention to this.

7. **Manage logistics, such as scheduling updates or preparing execution materials daily.** In addition, the business should facilitate data identification during testing, overall communication and any corrective action(s).

Stay Engaged in System Testing Cycles Prior to UAT

UAT represents the final and most important testing gate for a system implementation. However, the business cannot afford to disengage during previous system testing cycles, even though they are primarily the system integrator's responsibility. At minimum, the business should:

1. Verify that the overall test plan is comprehensive and covers all business requirements (i.e., tying the testing schedule to a requirements traceability matrix).
2. Achieve consensus among the process owners, the system integrator and the external auditor on documentation standards for testing.
3. Sample the testing documentation pre-emptively for each process, checking for:
    - Quality (Does the test scenario cover the business requirement comprehensively?)
    - Consistency of the recorded results with the supporting evidence (evidence must be retained)
    - Sign-off on test results by the appropriate stakeholders
4. Monitor progress by reviewing the tracker of open testing issues on a periodic basis.
5. Ensure regression testing is performed consistently to assess the impact of fixes for failed tests on test cases already passed.

Data Conversion

At a Glance

Business processes cannot be tested end-to-end without realistic data, and successful UAT hinges on data quality.

To avoid potential delays in the UAT cycle, the business should treat data conversion design and data cleansing as a top-priority work stream, not an afterthought.

To avoid data-specific issues that can create complications before and after go-live, the business must take operational and audit considerations into account from the outset of the system implementation project.

Configuring the new system to automate and optimise the firm's business processes is only half of the challenge. The other half is converting master, reference and transactional data into the eventual production environment. No two systems are alike, and data from one system will not map over directly or cleanly into a new system. Further, data quality issues in legacy systems can significantly delay the implementation program.

The data conversion aspect of an implementation is often ignored by the business, but its risks are as serious as those related to configuration. In particular, if the data conversion work stream is started too late, unanticipated system design and data design impacts might also be discovered too late. This can lead to a longer data conversion cycle, causing delays in building and validating the various implementation environments.

Moreover, since a business process cannot be tested end-to-end without realistic data, the successful completion of UAT depends on the quality of the data available in the system. Data conversion design and data cleansing are therefore essential to enabling UAT efforts and achieving confidence that the system will function in production as desired. System implementers often plan for a very limited role in data conversion beyond running tools to insert data into the new ERP system, and they typically leave the design, mapping, enrichment and cleansing activities to the client.

Again, if this critical work stream is treated as an afterthought, it can create data-specific issues that cause delays in the UAT cycle. It can also lead to work overload and time constraints for team members critical to testing activities. To avoid these complications, the business should take the following operational considerations into account:

1. In the early phase of the implementation program, clearly define the scope and strategy for data conversion. Secure sign-off from business owners of the data, as well as from the system integrator and the IT department.

2. Also during this phase, clearly define the design of data rules and validation criteria and secure sign-off from the stakeholders.

3. Assess the quality of the source data early in the program, as well. Create a plan that outlines clear responsibilities for cleansing and validating this source data. If data quality issues are identified for the first time later in the project life cycle, there is no shortcut for corrective action and the implementation program will be delayed.

4. Perform data validation multiple times to facilitate a smooth transition, or cutover, and go-live process.

5. Assess the timeline and resources required to convert and validate data. If possible, build specific tools to automate these procedures.

6. While it is generally preferred to check data populations completely, large data sets might require checksums or selective sampling. Assess these decisions diligently on a case-by-case basis.
7. To ensure the conversion process does not miss key data, establish an effective fallout (defect) management process. This process should include fallout handling procedures. These procedures become critical during cutover when fallout data must either be reloaded on short notice or added to a backlog to be addressed later in the production environment, post go-live.

Audit Considerations

Typically, the company's external auditor will be concerned about aspects of the system implementation that could potentially result in significant disclosure errors due to data integrity issues. Data conversion and system testing are primary focal points.

The auditor will seek to confirm that the business has tested data conversion comprehensively in the production environment as part of go-live. Given the potential severity of the impact of an unsuccessful conversion for the production environment, most organisations attempt to achieve this standard in one (UAT) or more (CRP) cycles prior to go-live. The external auditor will require:

- Formal sign-offs from management on the data conversion strategy and plan
- Data conversion designs and attribute-mapping documents in sufficient detail
- Validation procedures for each conversion object
- Validation of the conversion results for completeness and accuracy in both the pre-production (UAT) and production environments

In general, the auditor will select a sample of transactional objects – and potentially, master data objects – to perform detailed analysis of the data conversion validation and reconciliation. Typically, the system integrator is responsible for performing the data conversion. The system integrator must maintain an audit trail of the reconciliation between the source system files, the conversion files resulting from any manipulation of the source files, and the target system files.

The business, meanwhile, is responsible for managing the overall data conversion process by:

- Achieving consensus among the process owners, the system integrator and the external auditor on the format of the data conversion documentation and results
- Ensuring the completeness of the plan and the scope for the data validation process
- Training, guiding and supporting the process owners responsible for performing the validation
- Monitoring resolution of data conversion errors for both the UAT and the production conversion runs
- Performing a review of the validation results prior to handoff to the external auditor
- Developing documentation that shows that the company followed a sound and comprehensive data conversion process

Data Governance

At a Glance

The business must establish strong data governance that extends beyond successful rollout of the new system. It must also support data conversion and testing activities.

To ensure master data and transactional data are employed appropriately and consistently throughout the organisation, the business should develop a comprehensive data governance program that includes:

- A framework of organisational roles
- A "data dictionary"
- Defined metrics
- Documented policies

The business must establish strong data governance prior to go-live to support the data conversion and testing activities outlined above. This entails:

- Identifying and engaging subject-matter experts who can define acceptable data quality thresholds
- Establishing a data quality baseline (i.e., a measure of accuracy and completeness for existing data)
- Overseeing the data remediation process by establishing data quality reporting and prioritising the remediation efforts
- Identifying business owners for master data and ensuring completeness of business rules governing the creation and maintenance of master data
- Reviewing existing security policies and controls to determine their suitability to meet future business and regulatory requirements

Building a Comprehensive Data Governance Program

Data governance does not end with the successful rollout of a new system. Post go-live, the business must ensure that both master data and transactional data are employed appropriately and consistently throughout the organisation. The business can achieve this by establishing accountability for the accuracy, completeness and availability of data through a comprehensive data governance program. This entails the following steps:

1. Defining a framework of organisational roles and a program charter
2. Identifying data stewards who will own specific data objects and develop policies and procedures for managing the life cycle of their respective data
3. Documenting the life cycle of data (i.e., the business processes that govern how data is created, changed, deleted and archived) and defining an associated responsible/accountable/consulted/informed (RACI) model
4. Developing a "data dictionary" for master data structures that defines the business meaning of those structures as well as any business rules governing their attributes
5. Defining metrics that establish a baseline for data quality and can be used to drive data quality improvements
6. Documenting policies for securing and archiving data
7. Facilitating selection of software tools needed to support these data governance activities

Business Intelligence and Reporting

At a Glance

To ensure key reports are available immediately upon go-live of the system, and that the overall business intelligence environment is flexible in meeting both immediate and long-term needs, the business should:

- Plan for a complete, validated inventory of key reports in the implementation
- Review and validate the process for developing reports after go-live
- Review and validate the reporting architecture, tools and technologies
- Consider instituting a formal governance program to sustain the business intelligence environment

Establishing a robust reporting and analytics solution is often an afterthought in a system implementation. This results in business users struggling to piece together reports at the last minute, even with the new system in place.

Key reports should be available immediately upon go-live. A process should be in place to deliver additional reports post-implementation, and the overall business intelligence environment should be sufficiently flexible to meet long-term needs. The business can achieve these goals by:

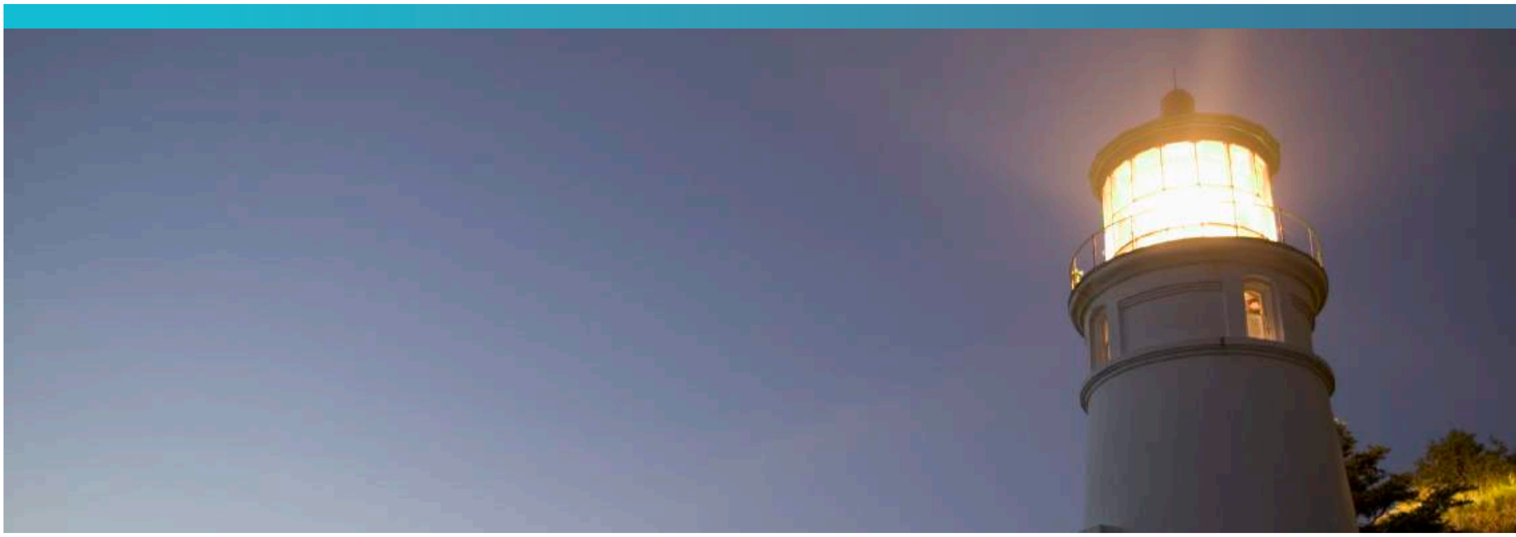
- Creating a complete inventory of key reports, verified by stakeholders, and planning for it in the implementation
- Reviewing and validating the process for developing additional reports immediately after go-live (months 1 to 3)
- Reviewing and validating the overall architecture and the tools or technologies in place for reporting to ensure they will meet future business needs

Lastly, the business should consider implementing a formal governance program to sustain the business intelligence environment. This program should include demand management and prioritisation so that the business can avoid report proliferation and ensure that reports generated align appropriately with business goals.

Conclusion

The role that the business plays in an ERP implementation is at least as critical as that played by IT and system integrators. Business leaders should understand that role, the choices to be made in how they prepare to execute it and the implications of choosing to delegate or deprioritise certain aspects of the role. This paper seeks to provide business leaders with a frame of reference in defining their role during an ERP implementation. No two ERP projects are exactly the same, and how an implementation will be organised will depend on a several factors, including past experience, partner methodologies and much more. However, all of the concepts presented here apply in one form or another.

ERP systems can bring remarkable efficiencies and return on investment to an organisation, or be a massive failure – and the business, not the integrator or IT, is ultimately responsible for the outcome.



HOW PROTIVITI HELPS COMPANIES SUCCEED

We use a business-driven approach with a goal of a high degree of process automation and effective analytics to help companies arrive at a solution (process, data, system) that meets the needs of their business at go-live. We seek to blend recommended process designs with the underlying automation capabilities of the ERP system. We provide strong governance practices and support drawn from prior experience with complex ERP projects. Our ERP experts bring technology, business process, data and program management skills to deliver efficient solutions across functional areas, ERP platforms and various implementation stages.

We help organisations manage their ERP implementations through a variety of partnership models. From advisory to workstream delivery to augmentation of a company's in-house teams, Protiviti brings expertise, intellectual property and leading industry tools to successfully deliver on key aspects of the ERP implementation:

- ERP Selection and Solution Design
- ERP Program Management
- Process and Solution Design
- UAT Design and Management
- Data Conversion and Governance
- Change Enablement and User Training
- Reporting and Analytics Design and Development

ABOUT PROTIVITI

Protiviti (www.protiviti.com) is a global consulting firm that delivers deep expertise, objective insights, a tailored approach and unparalleled collaboration to help leaders confidently face the future. Protiviti and our independent and locally owned Member Firms provide clients with consulting and managed solutions in finance, technology, operations, data, analytics, governance, risk and internal audit through our network of more than 85 offices in over 25 countries.

Named to the 2021 Fortune 100 Best Companies to Work For® list, Protiviti has served more than 60 percent of *Fortune* 1000 and 35 percent of *Fortune* Global 500 companies. The firm also works with smaller, growing companies, including those looking to go public, as well as with government agencies. Protiviti is a wholly owned subsidiary of Robert Half (NYSE: RHI). Founded in 1948, Robert Half is a member of the S&P 500 index.

CONTACTS

Steve Cabello
Managing Director, Los Angeles
+1.213.327.1470
steve.cabello@protiviti.com

John Harrison
Managing Director, Houston
+1.713.314.4996
john.harrison@protiviti.com

Toni Lastella
Managing Director, New York
+1.212.399.8602
toni.lastella@protiviti.com

Tammy Mittelstaedt
Managing Director, Chicago
+1.312.931.8719
tammy.mittelstaedt@protiviti.com

Martin Nash
Managing Director, Tampa
+1.813.348.3374
martin.nash@protiviti.com

Aric Quinones
Managing Director, Atlanta
+1.404.240.8376
aric.quinones@protiviti.com

THE AMERICAS

UNITED STATES
Alexandria, VA
Atlanta, GA
Austin, TX
Baltimore, MD
Boston, MA
Charlotte, NC
Chicago, IL
Cincinnati, OH
Cleveland, OH
Columbus, OH
Dallas, TX
Denver, CO
Ft. Lauderdale, FL
Houston, TX
Indianapolis, IN
Irvine, CA
Kansas City, KS
Los Angeles, CA
Milwaukee, WI
Minneapolis, MN
Nashville, TN
New York, NY
Orlando, FL
Philadelphia, PA
Phoenix, AZ
Pittsburgh, PA
Portland, OR
Richmond, VA
Sacramento, CA
Salt Lake City, UT
San Francisco, CA
San Jose, CA
Seattle, WA
Stamford, CT
St. Louis, MO
Tampa, FL
Washington, D.C.
Winchester, VA
Woodbridge, NJ

ARGENTINA*
Buenos Aires

BRAZIL*
Belo Horizonte*
Rio de Janeiro
São Paulo

CANADA
Toronto

CHILE*
Santiago

COLOMBIA*
Bogota

MEXICO*
Mexico City

PERU*
Lima

VENEZUELA*
Caracas

EUROPE, MIDDLE EAST & AFRICA

BULGARIA
Sofia

FRANCE
Paris

GERMANY
Berlin
Dusseldorf
Frankfurt
Munich

ITALY
Milan
Rome
Turin

THE NETHERLANDS
Amsterdam

SWITZERLAND
Zurich

UNITED KINGDOM
Birmingham
Bristol
Leeds
London
Manchester
Milton Keynes
Swindon

BAHRAIN*
Manama

KUWAIT*
Kuwait City

OMAN*
Muscat

QATAR*
Doha

SAUDI ARABIA*
Riyadh

UNITED ARAB EMIRATES*
Abu Dhabi
Dubai

EGYPT*
Cairo

SOUTH AFRICA *
Durban
Johannesburg

ASIA-PACIFIC

AUSTRALIA
Brisbane
Canberra
Melbourne
Sydney

CHINA
Beijing
Hong Kong
Shanghai
Shenzhen

INDIA*
Bengaluru
Chennai
Hyderabad
Kolkata
Mumbai
New Delhi

JAPAN
Osaka
Tokyo

SINGAPORE
Singapore

*MEMBER FIRM

Protiviti is not licensed or registered as a public accounting firm and does not issue opinions on financial statements or offer attestation services. PRO-0921-103090I-IZ-ENG