Part One Project management phases

1 Structuring an e-business project

When implementing an e-business project a number of processes and structures are required to ensure the project is successful. These include determining the correct structure to guide the course of the project, selecting the appropriate technology to implement, having the correct support technology in place to ensure the implementation will succeed, and choosing the right staff to carry out the project.

Many projects fail because these four elements have not been set up and employed correctly from the beginning. For example, the Standish Group conducted a survey of project failures with 365 organizations of all sizes across a wide range of major industry sectors. Focus groups and personal interviews were also conducted to provide a qualitative assessment of the survey.

Results of this survey showed that over 80 per cent of all projects suffered some form of failure. Only 16.2 per cent of surveyed projects were delivered within time and budget, while 52.7 per cent of projects were delivered but ran over budget, over time and had fewer features than were originally intended. Projects that were cancelled during their development formed 31.1 per cent of the sample.

Project failures included having to restart projects (94 per cent of all projects), cost overruns resulting in an average increase of 189 per cent of original cost, and time overruns, resulting in projects running an average of 222 per cent over original time estimates. Of all companies surveyed, an average of 61 per cent delivered the features originally specified.

The survey found key reasons projects were delayed or failed completely included lack of planning, low user input, incomplete or changing specification of requirements, lack of resources and competent staff, incompetence with technology, unrealistic timeframes and unclear objectives.

A later survey conducted in 2000 by the Standish Group found that 28 per cent of commercial projects were successfully delivered on time and budget with the required functionality. Of the remainder, 49 per cent suffered from partial failure and 23 per cent complete failure. In the government sector 24 per cent of projects succeeded while 50 per cent failed partially and 26 per cent failed completely.

The improvement in figures for project success between 1994 and 2001 was attributed to smaller projects being conducted, which have a higher likelihood of success.

Creating a successful e-business project therefore requires the project be planned correctly from the outset, structured into discrete project steps with identifiable and achievable goals, and the correct staff selected before the project begins.

The following sections discuss these issues, detailing the correct structure and process for conducting an e-business project, and the staff required for fulfilling the project.

1.1 E-business project management

The key to running a successful e-business project is to provide sufficient structure and planning to ensure project success. Success is typically defined as the project meeting its business requirements without running over budget or over time.

Therefore, the project structure should include a set of critical elements that govern the lifecycle of the project and its resulting outcomes. These elements follow the project from its inception to completion, and include the initial project planning, the requirements phase, the solution research phase, the design phase, the build phase, the pilot phase, the implementation phase and the project handover phase. Each phase should be completed with a corresponding set of documents that detail the information gathered in each phase, and the outputs each phase produced.

The initial project planning phase determines the broad outline of the project.

This is used to guide the initial project structure, including an outline of what the project is intended to deliver, and covers preliminary project planning issues such as potential technologies, budget, skills and timeframes.

The requirements phase extends the initial planning to determine the core set of deliverables the project should satisfy, which are in turn used to gauge project success. These cover all areas of business and technology relevant to the company, and are subject to analysis to prioritize the most relevant set of requirements to deliver with available resources.

The solution research phase is used to conduct detailed research into potential technology solutions to fulfil the initial project planning and requirements deliverables. This is then analysed and a set of potential technology solutions researched, with the best solution being recommended to proceed into the design phase.

The design phase takes the recommended solution from the research phase to create a high-level conceptual design for the solution. Following best practice, this design is then audited internally and externally to prove its feasibility, before a detailed design is created to cover the chosen technology solution in more detail, including application design, security systems, and the deployment configurations.

The build phase uses the results of the design phase to create the intended solution. Blocks of functionality are coded, deployed and tested using a build process across development, testing and production environments. The complete solution is iteratively assembled using this process by creating and integrating successive blocks of functionality until all chosen requirements are satisfied by the solution.

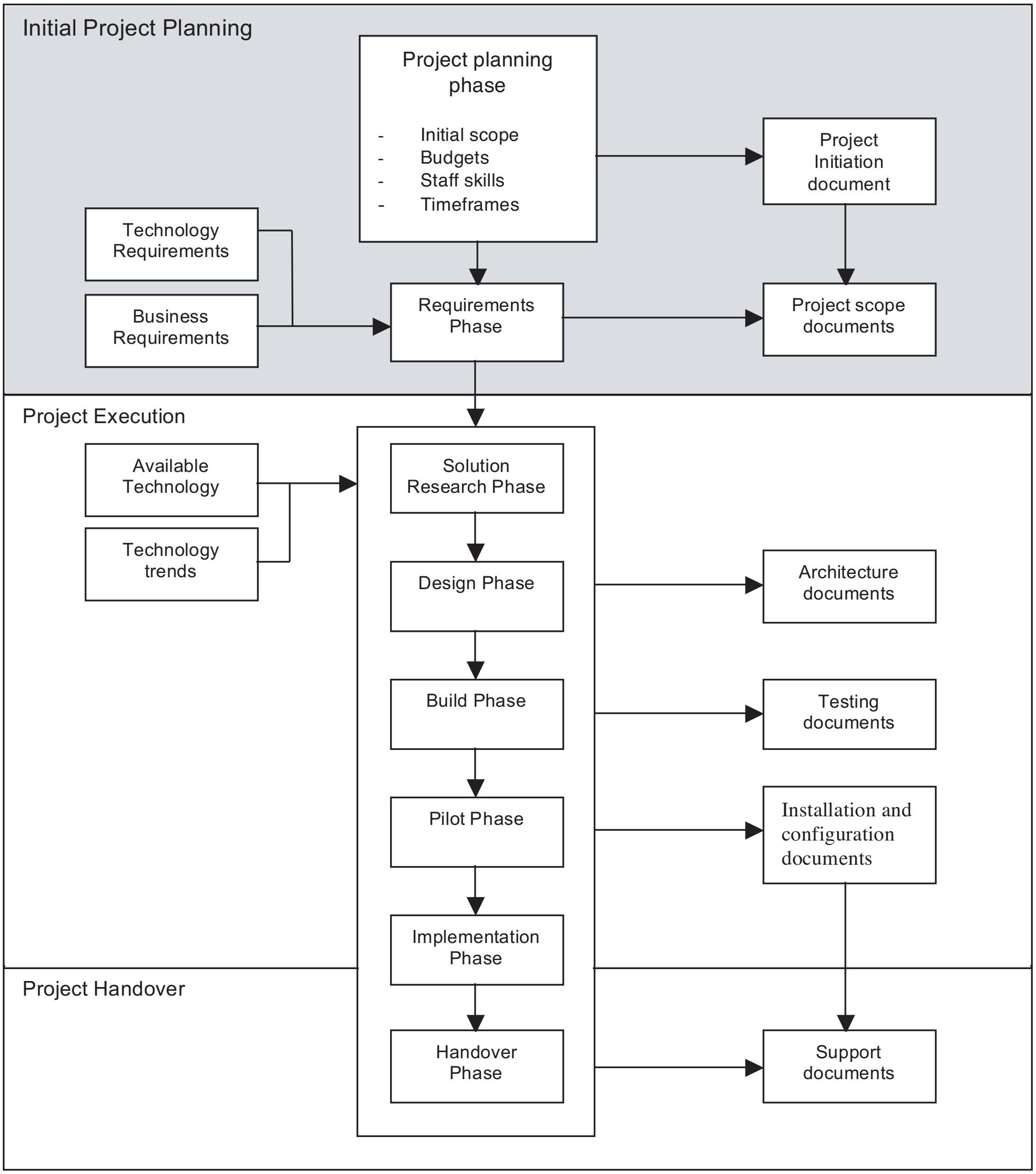
The pilot phase deploys an initial version of the completed solution for early testing by stakeholders (the members of staff or partners and suppliers who either sponsor a project or who are most affected by its implementation) and users. This allows the project to tune the solution to better fit requirements and satisfy deployment issues.

The implementation phase expands on the pilot phase to deploy the final solution across the business to all relevant business users. This requires the solution be deployed in a production capacity, capable of supporting all users and workloads, and the training and transition of users from old work practices to the new solution.

The handover phase collates the output of all previous phases into a set of support resources for operational and support staff. It also includes training of support staff, and the creation of final project documentation.

This structured approach to designing and running a project is depicted in Figure 1.1.

Figure 1.1 Structured project planning



This structured process begins with the initial project planning and requirements phases, and progresses through the main project execution phases until the completed project handover phase, where the project is turned over to internal teams for ongoing support. Each phase requires a set of inputs and produces a set of documented outputs that are required for each successive project lifecycle phase. These well-documented outputs are a fundamental advantage to this approach, and are used for project support, future developments, and auditing purposes.

The following sections discuss the elements of this structured approach to running e-business projects in detail.

1.2 The project planning phase

Before a project begins it is essential to have a broad understanding of the issues the project will address, and how the project will fulfil these issues. These are detailed in the project initiation document, which details the business problems facing the company, a broad overview of potential technology solutions, and estimates of the amount of time the project will take, what it will cost, and what staff will be required. These estimates are intended as a guide to assist in setting up the project, and are finalized in successive project phases.

An internal project manager and an internal or external e-business technical architect typically conduct the initial project planning, with each team member occupying distinct roles. The project manager is responsible for managing and tracking the project to ensure each phase is being delivered successfully. They typically create preliminary budgets and project timeframes in collaboration with the e-business technical architect. The e-business technical architect conducts preliminary research and assessments of the likely technologies necessary to solve the business problem.

The initial project planning is critical to establishing a clearly understood context for the decision-making, through a focus on why there is a need for the e-business project. This involves getting the different stakeholders affected by project decisions or involved in the decision-making process to understand and agree with the problems that are to be addressed. This is critical for 'buy-in' to the project, so that possible conflict between different stakeholders is prevented or minimized.

The core element of the initial project planning is the statement of the nature of the business problem confronting the company. This typically provides the motivation for the company to conduct the e-business project, and the extent to which the company resolves this problem determines the success of the project. It is therefore the focal point for understanding and determining intended technical solutions.

The statement of the business problems should include the nature of the business, the challenges facing it, and what business problems the technology is required to solve. It should also incorporate statements of the medium-term and long-term strategic views of business and technology needs. This ensures that all proposed solutions are aligned with medium-term and long-term plans, thus preventing selection of temporary solutions that will be discarded in future. The statement of the business problem should be expressed in a very simple and clear form, as shown in Figure 1.2.

Figure 1.2 Statement of business problem

Nature of Business

Company X is a financial services provider selling packaged pension, insurance and investment products to companies and direct to consumers.

Business Requirements

Company X's operating costs are increasing. They wish to restructure their business to use technology more efficiently to save costs and communicate more effectively with their own staff and their customers. They want their technology solution to reflect this.

The current problem
The company has a website and an Intranet site, and want to automate these to increase efficiency and create a cost-effective solution. They want to offer all their products online for self-service to clients, in real time, and utilize existing systems within the business including back-end legacy mainframe applications and their external suppliers of pension and investment products.

Alignment with medium-term strategy
Move to complete automation throughout their business. Enable integration of financial, HR systems, call centres, and marketing systems to provide real-time knowledge of all parts of their business.

Alignment with long-term strategy
They want to be able to provide customers with a real-time customized financial product portfolio that can be changed at any time depending on their circumstances.

Defining the business problem at this stage is also critically important as it provides a baseline to assist in minimizing the risks that may arise from making incorrect decisions during the course of the project. Frequent sources of risk include changes in business issues arising during the project, which in turn invalidate subsequent stages in the decision-making process, and potentially the technology solution selected and deployed.

However, these risks can be mitigated if the business issues have been made explicit at the outset of the project. Any changes in business issues or other requirements during the project can then be compared to the original assumptions, and if they differ significantly, the project can be modified to accommodate them without seriously jeopardizing the outcome.

Once the business problem has been stated, the e-business technical architect conducts preliminary research to provide prospective technology solutions and preliminary budgets and timeframes. Typically, this requires the e-business technical architect to research a range of appropriate design patterns and products from technologies commonly applied to specific business problems. These patterns typically include collections of technology architectures and products, design and development methodologies, and deployment factors utilized in previous e-business projects.

Selection of a range of appropriate design patterns and products is complicated due to the very large volume of technology information available, and the complexities and risks inherent in matching technology solutions to business problems. Therefore, this requires the e-business technical architect to have specialist knowledge, experience and skills of e-business technology, a broad and detailed understanding of the technology industry as a whole, and the ability to match technology solutions to business requirements.

The use of a formal research approach by the e-business technical architect provides several benefits to the project. It offers the ability to shorten the research phase, as the design patterns and products provide a structure to guide research. As the design patterns utilize proven pre-existing successful technology strategies, they lower the risk of making incorrect choices and allow the architect to reduce project timeframes. Finally, the use of such patterns shifts the solution focus away from individual point solutions reactively designed to solve a single business problem, and permits the e-business solution to be synchronized with long-term strategic planning within an enterprise.

This preliminary research must also consider a number of factors that are in turn related to the statement of the business problem. For example, the statement of business problem listed above would influence the technical architect to research e-business design patterns capable of targeting multiple channels, including online and offline channels, and supporting CRM functionality. They would also research solutions capable of integrating with financial services back-end systems, typically legacy AS/400 or mainframe products, and products that can be used with their partners and suppliers. Their choices should also be capable of extension to web services in the future, allowing for alignment of the solution to future medium-term and long-term strategic goals.

The factors affecting the preliminary decisions of the e-business technical architect can be classified into business-specific factors, and general external factors related to industry, vendor and technology trends.

Business factors

Business factors are comprised of business issues specific to the company that may affect initial technology choices. These typically include areas such as the company industry sector, available budgets, current technologies in use, and available staff skills and timeframes.

Industry sector factors are unique to each industry segment a company is active within. For example, manufacturing or retail businesses frequently require real-time information regarding stock levels, while service specific businesses may require a single complete view of all customer information for call centre staff.

Budgets provide limits to the purchase and implementation costs of technologies. They also constrain other resources employed during the project phases, such as skills and levels of staffing employed, and may also influence the availability of resources during later project stages, such as the duration and degree of testing in the build phase.

Initial project planning is also influenced by the technologies currently utilized within the business, due to the large amount of investment a company may already have made in specific areas, and a desire to lower support and administration costs through the reuse of infrastructure. These technologies typically include hardware systems, operating systems, peripherals, and enterprise applications such as business productivity, enterprise resource planning, and e-business systems.

The presence of existing technologies within the business is frequently viewed as an important business factor, and is often given greater weight than critical factors such as the performance or availability of the solution. Overemphasis of this factor may result in the technical architect selecting compatible technologies with resulting tradeoffs, such as increased project timeframes, higher total project costs, and lower levels of reliability, scalability and availability in the final solution.

For example, a company may require a highly available e-business solution to ensure customer satisfaction and maintain high customer retention rates. However, they may have standardized on a particular set of supported technologies that are unreliable for e-business purposes. If the e-business system must conform to these technology requirements, higher levels of system downtime may jeopardize customer satisfaction, resulting in lost business. This may in turn considerably outweigh the savings made from using common but unreliable infrastructure. Therefore, the requirement for the adoption of a more reliable technology should be preferred over selection of common systems.

The mixture of skills available within the company's information technology department may also help to determine what type of technology should be selected, as the department may already have made a significant investment in skills development in this area. For example, if an information technology department has the majority of its development skills in one area, selecting that development technology may be a business factor. However, in a similar manner to the existing technologies within the business, levels of internal skills should not outweigh critical requirements such as the ability to create a suitable e-business solution.

Finally, the total time available for a given project will have a strong influence on the initial project planning. Projects that must deliver functionality within short timeframes will influence decisions towards pre-packaged off-the-shelf technologies capable of meeting business requirements with minimal customization. If a project is allocated long timeframes, decisions can include technologies requiring more custom development. However, long duration projects of over 12 months are not recommended as they frequently lead to cost and time overruns, and result in technological obsolescence.

External factors

External factors are issues outside the business that may influence the initial project planning technology choices. These factors include existing relationships with technology vendors, technologies used by external partners and suppliers, trends within the information technology industry, and technology adoption trends within other business segments.

Frequently a company will have existing relationships with multiple technology vendors. Selecting solutions from a vendor with whom the company has an existing relationship and direct previous project experience can provide a significant reduction in the risks of conducting a project, provided this relationship is close and beneficial to all parties. However, this factor must be balanced against the need to select the vendor technology solution most appropriate to solving the business problem.

Integration of systems with external suppliers of products and services will also influence technology decisions. For example, if a supplier has a proprietary enterprise application integration solution, potential integration technology choices with that supplier may be restricted. Such relationships will also influence the choice of other related technologies, such as security systems, to ensure such close integration will not compromise corporate systems and information.

However, selecting potential technology solutions that are tightly coupled to specific suppliers may increase project risk. This occurs when the solution compromises other internal business factors that must be addressed in the project, and inhibits support for and integration with additional suppliers and future technologies.

Technology industry trends frequently influence the selection of potential technologies due to the expected lifecycle of a technology. If a technology is becoming obsolete, it makes less sense to consider selecting it for a future project. If a technology is becoming more widespread, it may be sensible to utilize it if many vendors will support it. Similarly, technology selections are often influenced by the trend towards adoption of packaged off-the-shelf technology solutions rather than creating proprietary customized solutions.

Finally, trends adopted by competitors also influence preliminary technology decision-making. For example, when many businesses in an industry sector employ the same or similar technologies, companies frequently give these more importance in making initial technology decisions. However, this strategy should be avoided, as each technology should be assessed on its business merits and degree of fit for the company and issues at hand. Alternatively, analysis of how other companies employ technology can indicate the technologies to avoid if there are widespread problems in that industry segment, or how to correctly deploy such technologies to avoid common mistakes.

Following determination of the internal and external factors and their influence on preliminary technology decisions, the e-business technical architect creates an overview of the technology solution proposed to address the business problem. This will typically include a broad overview of the technology functionality required to satisfy the business problem, and the business and external factors. It will also typically include at least three potential solutions from different vendors, and provide an indicative costing for each alternative.

For example, with the example depicted in Figure 1.2, the technical architect's overview may include three proposed solutions for the creation of an internal and external e-business portal solution, integrated with a transactional e-business for customers and internal legacy and partner and supplier systems. These may include specific vendor products and indications of proposed customizations and content development required, and deployment scenarios.

The project initiation document is then created, including the statement of the business problems, the proposed preliminary technology solutions, and estimated budgets, staff skills and project timeframes. This forms the first output generated in the course of the project.

This document should be read and approved by the business and technical stakeholders within the company, as it provides the context for the major decisions within the project. It also provides a mechanism for initial agreement on how the project will be run and what it will deliver. Once this document has been agreed with stakeholders, the project can move into the requirements gathering phase.

1.3 The requirements gathering phase

The business issues introduced in the initial project planning phase provide a general context to the business problem that must be solved. This context must be extended during the requirements gathering phase to provide a more detailed description of all the business and technical issues the project must provide a solution for. These are in turn detailed within the project scope document, which forms the second project output.

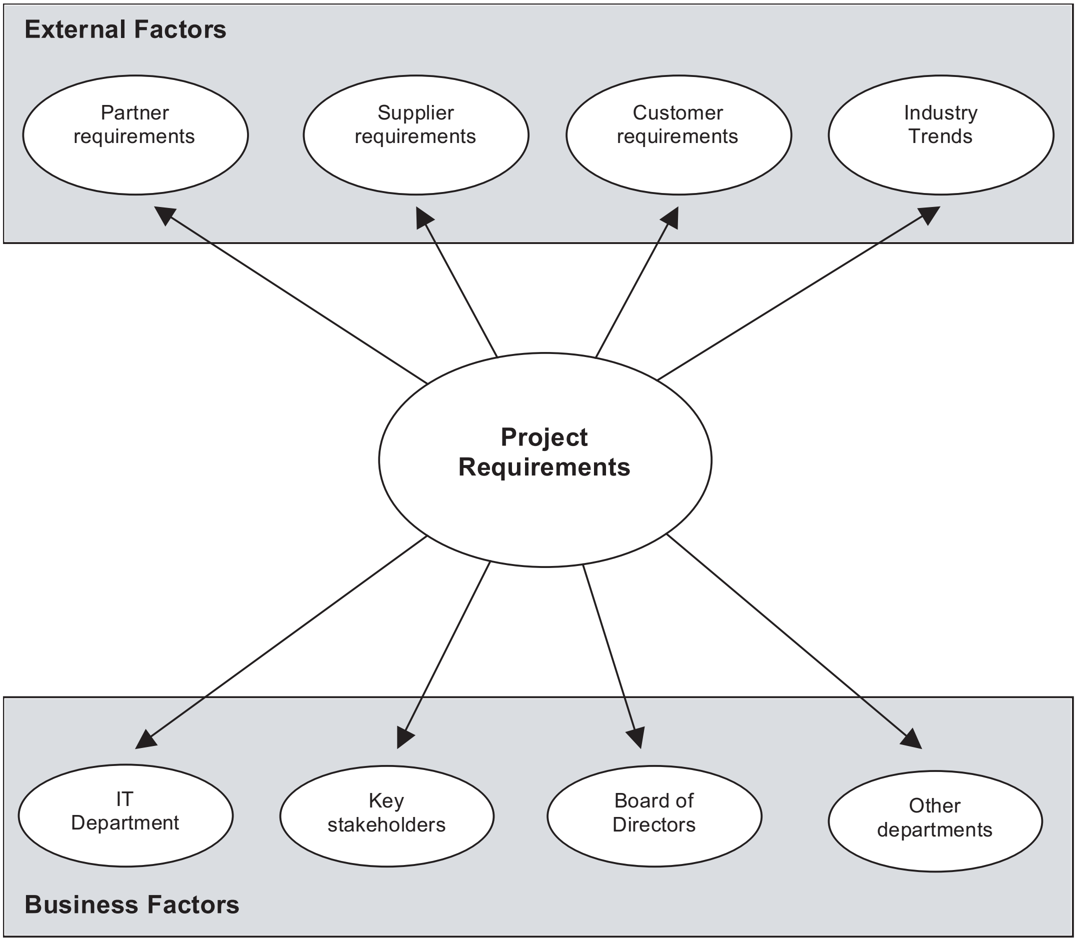
During this phase, the project manager is responsible for overseeing the gathering and analysis of the requirements. They must also ensure that appropriate business analysts are hired, and ensure they work closely with the e-business technical architect and business stakeholders. The business analysts are responsible for gathering and analysing the business and technical requirements from business stakeholders and customers using one or more of the methods described below. They must also work closely with the e-business technical architect to ensure the technical issues specific to the company and project are addressed in full. The e-business technical architect is also responsible for analysing these requirements before the subsequent research and design phases, which are used to create the high-level and detailed designs.

Project requirements are the detailed issues that combine to create the initial business problem. These are typically composed of necessary business and technical conditions within the company, or problems experienced with specific business processes by stakeholders. Each requirement imposes specific constraints on the project solution, and must therefore be addressed by the project to ensure project success.

For example, a stakeholder such as an investment manager may use a manual process to assist a customer in choosing an investment product. Customers may wish to have this process automated in a convenient online system, thus creating the requirement for automated product selection using an e-business system.

Detailed requirements are obtained from internal and external sources, including internal company divisions, and external customers and partners and suppliers, and are categorized into detailed business requirements and technical requirements, as shown in Figure 1.3.

Figure 1.3 Common sources of project requirements



Detailed business and technical requirements originate from the external and business factors highlighted in the initial project planning phase. Once these sources have been identified, their specific requirements are gathered using a range of mechanisms appropriate to the company and business in question. These may include structured questionnaires and interviews with stakeholders, internal reports, strategy documents, and analyst reports.

The methods used to gather requirements will depend on the size of the project and the degree to which it will affect the business. Large and strategically important projects are typically driven from higher levels within the company, often at board level via the chief technology officer or chief information officer. Due to their strategic importance, such projects may require wide consultation within the company, and hence a formal and detailed requirements gathering exercise. This may utilize mechanisms such as published board directives and strategy reports, followed by interviews with stakeholders that are subsequently used to create detailed questionnaires. Such exercises often result in a very detailed and focused set of requirements.

In contrast, small projects with limited budgets may require rapid and more informal requirements gathering exercises. This may include interviews with critical stakeholders, and reference internally published reports and reports from IT analysis firms to gauge 'industry best practice' as a mechanism to shorten the requirements gathering phase.

Structured questionnaires are frequently used to obtain requirements from large numbers of company stakeholders, and typically solicit responses to items such as lists of business objectives, problems with business processes, and proposed solutions. Questionnaires are administered to all stakeholders across company business units, then collated into common problems and desired solutions. Information technology departments should also be included in such questionnaires, as they are normally required to support the solution once it has been implemented and can also contribute relevant technology and business process issues.

Interviews with stakeholders provide a less structured mechanism to gather requirements compared to questionnaires. However, interviews frequently produce very detailed sets of requirements that may be overlooked in questionnaires, such as revealing weaknesses in specific areas within business processes and tools. In addition, interviews with stakeholders may also be employed to create detailed targeted questionnaires for use in widespread requirements gathering throughout the company.

Interviews and questionnaires should also include external stakeholders, such as representative samples of customers and members of partner and supplier companies. Such stakeholders can provide feedback on end-user functionality requirements, and may contribute valuable high-level technical requirements such as desired levels of availability and performance for an e-business system. Requirements may also be gathered through internal reports and strategy documents, including high-level company strategy reports, and business unit documents such as policy and procedure documents. These typically provide short-term, medium-term and long-term goals; objectives and strategies that the proposed solution must satisfy, and may provide critical success factors and performance indicators, such as business improvement targets.

Strategy-based requirements may also originate from the board of directors of a company, which frequently establish broad goals such as ‘becoming an e-business leader’ in their industry field or ‘implementing a supplier trading website with partners’. Due to the pivotal and central place of technology within modern business, technology departments should be represented at board level via CIO or CTO members, and hence provide input into these strategies.

Additional sources of requirements gathering may include specialist analysis firms such as Butler, Forrester or the Gartner Group. Such firms often release reports from in-house analysts detailing the technical and business issues facing companies within specific business segments. These may provide valuable input into the definition of both technical and business requirements.

In general, complex projects that will have a deep impact within the company will require formal and widespread requirements gathering processes. Typically, such projects utilize custom developments to meet these requirements, and utilize formal requirements gathering and description methodologies such as the use case method used in Object Oriented software design. Use cases detail the set of actions and expected responses for each requirement, and are employed for rapid prototyping and simplified development cycles. In contrast, less formal and simpler projects require less involved requirements gathering processes, and typically express the resulting requirements in standard document templates.

Analyse requirements

Once the requirements have been gathered and documented, the project manager, business analysts and the e-business technical architect conduct a requirements analysis using a structured approach. This allows the requirements to be prioritized according to importance to ensure the project can provide the greatest degree of functionality within the available resources. It also provides a mechanism to clarify complex interactions between conflicting requirements by simplifying the amount of detail gathered in the requirements gathering phase.

Requirements are analysed through a process of ranking according to their significance to the project, and the level of risk they represent to the project. Significance is typically determined by the extent each requirement influences decision-making, and thus how critical they are to the final solution. Requirements that strongly constrain a project but threaten project success should be reassessed and modified or discarded if appropriate.

Requirements with high levels of significance frequently include project deadlines, requirement to reuse existing technologies, project budgets, and specific aspects of functionality such as support for payment methods, transaction types or specific customer-driven requirements.

Requirements with lower levels of significance typically include utilizing products from a preferred vendor or consultancy, or from in-house project resources such as existing development and support staff with particular skills.

Ranking of requirements is also affected by less tangible factors. These include factors such as industry trends, competitor trends and corporate preferences in specific technology areas. Such factors should be accounted for when prioritizing requirements, but they do not directly constrain decisions or carry high levels of risk to the project, and can therefore be allocated lower significance levels.

Sources of risk can be determined during the requirements gathering phase through interviews with stakeholders. Risk factors are typically expressed as the threat of the project failing from not satisfying a given requirement, and the specific elements that contribute to risk for that requirement. Formal risk management processes should be incorporated throughout each stage of the project to plan for risks that may arise, using a process of identifying and analysing risks, determining methods to handle risks, and providing ongoing monitoring of risks throughout the project. Risk management typically requires maintenance of a risk register by the project manager, in the form of a database, throughout the project lifecycle.

Once the significance and risk profile of each requirement have been classified, they are placed into an analysis matrix. This locates each requirement within rows and columns corresponding to the source of the requirement (business or external), the description of the requirement, the significance of the requirement, the risk it represents to the project if it is not fulfilled, and details of factors contributing to this risk.

An example of such a requirements matrix would typically include the elements depicted in Table 1.1.

Table 1.1 Requirements analysis matrix

Source	Requirement description	Significance	Risk	Risk Issues
Business	Must be delivered in 6 months	1	High	Must beat competitor to market
	Must integrate with partner X	1	High	They supply some critical financial products
	Budget capped at $3 000 000	1	Medium	Project very important therefore budget can grow
	Integrate with internal CICS financial system	1	High	Core products sourced from this system
	Strategic goal to adopt industry standard technology to lower costs	1	Low	Current trends support this form of development
	Availability	1	High	Must be continually available to service customer requests
	Scalability	1	High	Must account for varying levels of customer demand
	Security	1	High	Must prevent compromise of solution from internal and external sources
	Corporate guideline to use Unix based solutions	2	Low	Extensive in-house support and widespread industry support for Unix products
	Corporate hardware standard is for Sun SPARC systems	2	Low	As above
	Internal skills in Unix, SPARC, and COM technologies	2	Low	Can outsource development and support skills or retrain staff
	Integration with existing COM e-business system desirable	3	Low	Will not affect project launch date
External	Technology industry trends (list)	2	Low	Industry standardizing on Java e-business technology
	Current vendor relationships (list)	3	Low	Can readily utilize other companies
	Current supplier relationships (list)	3	Low	Can readily utilize other companies
	Competitor trends (list)	4	Low	Competitors utilizing similar technology

The requirements analysis matrix is included as the final section of the project scope document. It should incorporate enough detailed information to permit the e-business technical architect to begin the design phase of the project.

An example scope document is depicted in Figure 1.4.

Figure 1.4 Decision output: project scope document

Project Scope Document (title of document)

Table of Contents (list of all document headers)

Introduction
Briefly state the purpose of the document, and summarize the contents.

Objectives
State what the document is intended to achieve, i.e. the need to resolve the business problems being faced.

Document History
As this document will change over time, include version information in a table here with document author and the reasons for making changes, e.g. adding a new section.

Related Documents
In a table, list any other documents that are referenced within this document. E.g. published reports used.

Overview
State the key issues facing the business, why these require a technology decision, and how this will solve the issues.

Business Requirements
Discuss the source of the business requirements, how they were gathered, and discuss each requirement in logical groupings (e.g. business and external factors).

Technology Requirements
Discuss the source of the technology requirements, how they were gathered, and discuss each requirement in logical groupings (e.g. internal and external factors).

Requirements Analysis
Place each requirement into matrix of source, name, influence type, importance and risk. Include verbal summary of results of this analysis

Source	**Requirement description**	**Significance**	**Risk**	**Risk Issues**
Business	Must be delivered in 6 months	1	High	Must beat competitor to market
	Must integrate with partner X	1	High	They supply some critical financial products
	Budget capped at $3 000 000	1	Medium	Project very important therefore budget can grow
	Integrate with internal CICS financial system	1	High	Core products sourced from this system
	Strategic goal to adopt industry standard technology to lower costs	1	Low	Current trends support this form of development
	Availability	1	High	Must be continually available to service customer requests
	Scalability	1	High	Must account for varying levels of customer demand
	Security	1	High	Must prevent compromise of solution from internal and external sources
	Corporate guideline to use Unix based solutions	2	Low	Extensive in-house support and widespread industry support for Unix products
	Corporate hardware standard is for Sun SPARC systems	2	Low	As above
	Internal skills in Unix, SPARC, and COM technologies	2	Low	Can outsource development and support skills or retrain staff
	Integration with existing COM e-business system desirable	3	Low	Will not affect project launch date
External	Technology industry trends (list)	2	Low	Industry standardizing on Java e-business technology
	Current vendor relationships (list)	3	Low	Can readily utilize other companies
	Current supplier relationships (list)	3	Low	Can readily utilize other companies
	Competitor trends (list)	4	Low	Competitors utilizing similar technology

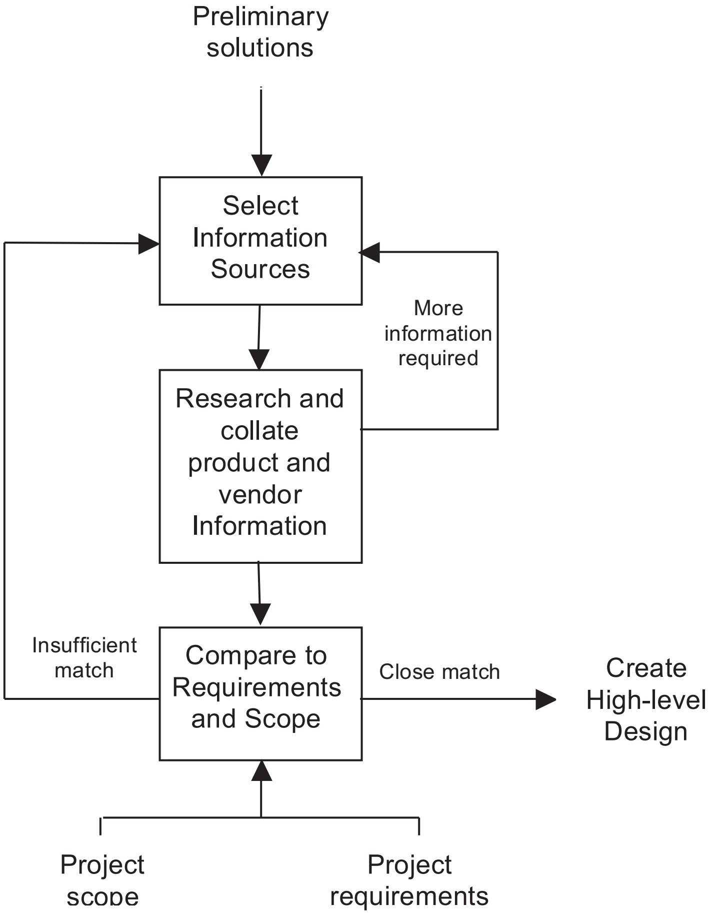
Summary
Summarize the findings of the document.

1.4 The solution research phase

Once the business and technical requirements have been determined, the solution research phase is conducted to produce a set of recommended technical solutions to solve the project requirements. The preferred solution then forms the basis of the proposed high-level design.

The e-business technical architect conducts the solution research phase. This involves a process of selecting sources of information about proposed solutions, conducting research and compiling information on the technology products and their vendors using these sources, and evaluating the results of this research against the initial scope and project requirements. This process is depicted in Figure 1.5.

Figure 1.5 Technology solution research process


Preliminary solutions
Select Information Sources
More information required
Research and collate product and vendor Information
Insufficient match
Compare to Requirements and Scope
Close match
Create High-level Design
Project scope
Project requirements

The e-business technical architect determines appropriate information sources for research based on the three initial technology solution overviews created in the project initiation phase. Typical sources for research include print and Internet magazines, print media such as trade publications in relevant areas, vendor websites, and market research reports.

Solution information is gathered and collated from these sources, including core functionality, product technologies and deployment options, and the product architecture. Analysis of this information by the e-business technical architect will typically employ multiple analysis techniques gained within previous projects. These include methods such as function point analysis comparing product functions to required functions from the requirements matrix and scope documents, analysis of vendor and industry analyst evaluations, assessment of user reports, previous experience with products, and the vendor factors. If a solution does not provide sufficient functionality, additional information must be gathered, or alternatively the solution must be discarded and a new solution selected and researched.

The process of technology solution research and selection is one of the most important decisions made during the project lifecycle, as it is crucial to project success to ensure the correct technologies are selected. E-business technology continually changes, and therefore selecting the wrong technology may result in systems that fail to work, or provide only a partial solution to business problems.

Technology selection should also account for existing corporate business and technology strategy, and other projects and technologies deployed within the company, to ensure that solutions can be integrated into other corporate systems. Appropriate selection also minimizes ‘orphaned’ solutions where technology is deployed that rapidly becomes obsolete and unsupported by vendors, resulting in lack of support and difficulty in obtaining product updates, and therefore increasing the operational cost to the business.

Following this process, the e-business technical architect should produce a final detailed recommendation for the proposed technology solution, and two alternative recommendations. Each recommended solution should be ranked according to the extent to which they fit the project scope and requirements. If the solution research phase cannot clearly distinguish between proposed alternatives, it is recommended that a small-scale proof of concept project be conducted using the recommended technology solutions to allow the e-business technical architect to more accurately determine the most appropriate solution.

Vendor selection

The role of solution vendors is an important element in product research, selection and resourcing for an e-business project. Vendor suitability for a project is determined by a set of factors, including suitability of products and services offered by the vendor for the project requirements, the viability and history of the solution and vendor, the cost of their solution and ongoing support, and the vendor's levels of customer service.

The suitability of vendor products and services to the business and technical requirements of the project is a key factor in ensuring project success. Selection of inappropriate vendor solutions is a common problem, and frequently results in a solution incapable of supplying the required functionality, necessitating additional and unexpected custom work to fit the solution to the project requirements. This also frequently results in increased costs and time overruns for projects, and requires high levels of support for the incorrect solution. It is therefore important to choose a suitable vendor and solution that closely matches project requirements, and has minimal need for customization.

Vendors should also be assessed to determine their ongoing viability over the following years in order to provide continued support, bug fixes and new features for current and outdated products. This ensures the solution can generate a reasonable return on investment before its replacement with newer, upgraded systems. Vendor viability typically includes the financial viability of the vendor, with a focus on selecting vendors and products with high or increasing market share to ensure their products will not be discontinued.

The viability of a solution can also be increased if vendors have widely available and well-supported products. Such products are sold and supported by multiple independent vendors, with consultants specializing in the product widely available from vendors and independent contractors. In addition, training in such products is typically readily available for internal staff. These factors allow companies to switch vendors to find suitable service and support levels, which in turn decreases vendor lock-in and reduces the risks to the project.

The strategic viability of products and vendors should also be considered. Vendors should demonstrate a history of innovation in e-business, and articulate a clear strategic direction for their products, including adoption of emerging technologies and standards, and targeting of specific market segments with appropriate features. This direction should also be aligned with the company strategic direction to ensure compatibility.

A vendor's strategic vision should also remain consistent over time and focus on providing business value for customers. Vendors attempting to profit at the expense of customers through frequent strategic shifts, licence changes, or forced product upgrades through incompatible product features should be avoided.

Once a product offering the required features has been selected from a viable vendor, the cost of solution should be determined. Typically, up-front purchase costs represent only a small percentage of the total cost of the solution. It is therefore imperative to determine the complete lifecycle cost of the solution from initial purchase to eventual replacement. Lifecycle costs include factors such as the ongoing cost of product maintenance, support and upgrades, security costs, training costs, and the usability of the solution, which affects employee productivity and customer satisfaction.

Reputable vendors should also be able to provide a detailed breakdown of their solution costs during early negotiations. This allows for price, performance and feature comparisons between competing products. Vendors also frequently offer discounts for start-up companies, the purchase of additional products and services, global licensing of products across multiple business units in large companies, and for companies subscribing to development programmes. Therefore, when obtaining quotes from vendors, companies should capitalize on such incentive programmes to ensure they obtain the best price for the chosen solution.

When evaluating vendors and vendor solutions, it is also recommended that trial copies of products be obtained either during the initial negotiations with the vendors, or from website downloads. This allows a company to conduct tests of features and integration capabilities, and assists in the determination of cost factors such as security, support, and usability. Availability of trial copies may also indicate vendor confidence in their products, and a willingness to build market share by seeding the market with product knowledge and skills.

Finally, the level of customer service provided by the vendor should also be considered during vendor selection. This includes factors such as providing professional service at all times, the sales experience for prospective customers matching the level of their technical after-sales support, vendors returning calls promptly and providing sufficient information to assist in decision-making (including knowledgeable technical staff), and providing guaranteed delivery dates for products and services. Such dates are critical during project execution to ensure valuable staff members do not have to wait for vendors to deliver project resources. Similarly, it is recommended that all dealings with vendors be conducted with equivalent levels of professionalism from the company to foster a good long-term working relationship.

1.5 The high-level design phase

Once the solution research phase has finalized a recommended technology solution the project progresses to the creation of the high-level design phase using the results of this research. The high-level design provides a formal overview of the technical components of the solution necessary to meet the required business and technical functionality specified in the requirements analysis.

The e-business technical architect is responsible for the creation of the high-level design. This occurs through analysis of the recommended solution to produce a set of functional and operational design components satisfying the business and technical requirements of the project.

High-level functional design components consist of software solutions that will be used to provide discrete groups of business functionality corresponding to stakeholder requirements. For example, a high-level functional design for a transactional e-business system may consist of a set of software components providing data storage, catalogue management, transaction processing, payment processing, interface management, and security. Each component is an independent entity that offers services to other components, with the complete set of components satisfying the business and technical requirements.

The appropriate high-level functional components are determined through an analysis of the project requirements, the interactions users will have with the e-business system and the features and systems offered in the proposed technology solution. Functional components may be provided through purchase of packaged solutions, or development of the appropriate components.

The operational components of the high-level design describe the physical aspects that govern how the functional design will be deployed, and are selected during the solution research phase. These typically include operating systems and hardware servers, development tools, network and security systems, and performance and availability levels. For example, the transactional e-business system described above may require an operational design consisting of specific hardware servers, operating systems, and network and security devices assembled in a network configuration to provide high availability and performance. Typically, the operational servers are used to host the functional design components.

Operational components are also determined from further analysis of technical and business project requirements, and from the functional design, which may indicate the need for additional systems.

Both operational and functional components may be simplified during the solution research phase by selecting solutions supporting multiple operating systems and/or development languages. This removes potentially restrictive dependencies between design components, such as specific products selected to provide functional components requiring specific operating systems. Removing such dependencies in turn allows the e-business technical architect more scope to optimize the functional design to suit requirements, and the operational design to support different deployment, performance and availability requirements.

Once analysis of the functional and operational design components is complete, they are included in the high-level design document. This document is intended to provide an overview of the design for internal and external audit, and to guide the subsequent creation of the detailed design and build phase.

The high-level design should therefore be written in a manner suitable for a technical and non-technical audience, and provide sufficient detail to justify the choice of design components while avoiding detailed discussions of implementation-specific issues such as detailed functionality or configuration and deployment information. Once complete, the high level design allows the e-business technical architect and project manager to finalize the project budgets and timeframes according to the chosen solutions and design.

The audit phase

Following completion of the high-level design phase, it is strongly recommended that the high-level design document be submitted to an external firm for analysis. This provides an independent audit for the design to ensure each requirement has been met, and to minimize the risk of making incorrect decisions. Specialist analysis firms such as the Butler Group or Gartner are recommended for this function, or alternatively another external e-business technical architect.

In addition, an internal audit of the high-level design is also recommended to gain stakeholder buy-in and ensure the proposed solution will be acceptable to relevant stakeholders. This internal audit also provides stakeholders with necessary feedback from internal staff to ensure their business requirements will be met.

1.6 The detailed design phase

Once the high-level design has been audited and approved, the detailed design can be created by the e-business technical architect. This design provides a detailed set of specifications of the functional and operational components of the design, which are then used by developers, implementation staff and web designers to create and test the system during the build phase.

The detailed functional design provides an in-depth description of the functional software components (known as the software component model) providing the functionality specified in the requirements document. This includes descriptions of the software components, the interactions between components (which components use other components), their software interfaces (their expected inputs and outputs), and descriptions of the externally observable behaviour of the components, typically described using 'use cases' and 'collaborations' created from the requirements analysis. Use cases specify how an external individual or system would interact with the solution, for example how a user would purchase a product from an e-business site. Collaborations specify the interactions between components over time, and are used to express the dynamic behaviour of the functional components. Each element of the detailed functional design is typically expressed using the Unified Modelling Language (UML) system of notation, providing a common language for the project team.

Functional components are structured through analysis of the required functionality and selected software solutions. The e-business technical architect assigns specific units of functionality to discrete software components using a range of design principles. These include distributing functionality between components into distinct layers, including presentation user interfaces, business logic and data access, and internal and external integration systems. It also requires avoiding redundancy and duplication in components, the incorporation of legacy systems and other corporate applications, and consideration of operational service level requirements such as performance, security and availability.

This analysis should produce clean separation between the application business logic, presentation, event management, data access, and integration components. This separation in turn permits the different groups of project staff to work on separate components in parallel during the course of the project, thus decreasing the time taken to create the solution. In addition, it permits each component to be modified separately from the others, allowing for rapid changes during the build phase such as the addition or alteration of solutions features. It should be noted that in addition to in-house development of functional components, much of the required functionality may be purchased as packaged solutions and integrated with custom project-specific code.

The detailed functional design components are also used to create test cases, which consist of specific sequences of steps and their related conditions used to test the expected behaviour of functional components. Creating test cases early in the build phase ensures that testing is closely aligned to the development of the solution, and allows for rapid and interactive modifications to systems to fulfil the specified requirements.

The detailed operational design consists of a set of components responsible for how the system will be physically deployed, including specific products such as hardware servers, network devices, security devices, and the physical placement of these components within the company. These components are typically expressed using network structure diagrams, depicting the location and connections between operational components, the functional components deployed across these, and specific configuration documentation.

In addition, the operational design includes non-functional issues such as the performance, availability, and security of the solution, support processes and procedures for the ongoing operation of the solution, and internal and external technology requirements such as preferred operating systems. These aspects of the design are typically expressed through service level agreements (SLAs), and support documents such as installation and configuration documents.

The detailed operational design must be synchronized with the functional design to ensure the design is capable of satisfying the non-functional aspects of the design, such as performance. This requires the e-business technical architect to analyse the flows of data within the solution according to user activity and data source, the deployment configurations of packaged products, and the non-functional operational requirements. This process should occur simultaneously with the functional design to ensure that operational design factors are built in to the complete solution from the outset.

The detailed design must also specify project change control mechanisms. These include software code and document management products such as the Open Source CVS, the Rational Clear Case product, or Microsoft Visual

Studio/SourceSafe. Such tools provide a central repository for all project code, and can enforce rigorous change control of functional component source code to ensure the correct code is utilized and deployed during the build phase. In addition, change control procedures should also be specified for operational systems (such as server configurations), and project documentation. This provides an audit trail for changes made during the project, and ensures that arbitrary, unnecessary and potentially problematic changes are not enacted by project or company staff.

The resulting detailed design provides a blueprint for the creation of the solution in the build phase. However, it should include sufficient flexibility to allow for modifications during the build phase, such as restructuring the deployment of functional components across operational systems for performance. It should also support future enhancements as requirements evolve, such as the addition of new components or modifications to existing components.

1.7 Build phase

Once the detailed design has been approved, the build phase begins. This phase involves creating a working solution to meet the project requirements, using the specification supplied by the detailed design document.

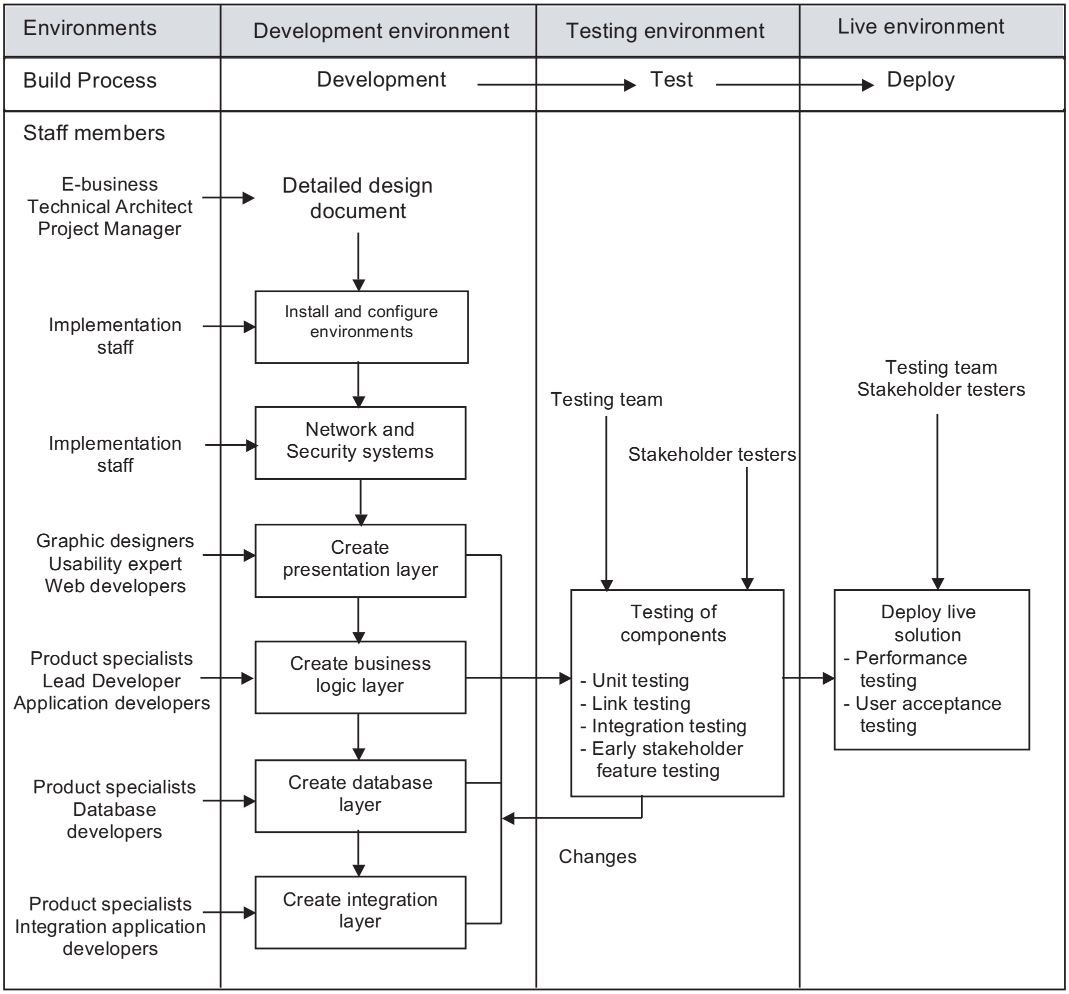
During the build phase, the e-business technical architect ensures that the development and testing of a production-ready solution is managed and implemented correctly according to the detailed design document. This includes tracking technical milestones to ensure design components are correctly delivered on time, ensuring the performance, reliability and security of the complete solution is maintained, and enforcing technical change management to minimize errors during development.

Similarly, the project manager oversees the build phase as part of the ongoing management of the project. This includes ensuring staffing, resourcing and client issues are managed, that project milestones are met, and that project risk issues are addressed as they arise.

The build phase begins with implementation staff members skilled in the appropriate technologies and products installing the detailed operational components of the design into the development, testing and live production environments before application development commences. These include components such as web servers, application servers and packaged business logic products. It also includes source code control servers used to store the software created by the development team in a central repository, and computers and tools used by members of the development teams to code the business logic, databases, integration, and presentation layer systems. In addition, network and security systems are installed across the design layers, and consist of network hardware such as switches and routers, and security systems such as firewalls, intrusion detection systems, and server hardening procedures.

The development, test and live environments, build phase tasks, and staff members responsible for these tasks are depicted in Figure 1.6.

Figure 1.6 Environments and processes used during the build phase


Environments
Development environment
Testing environment
Live environment
Build Process
Development
Test
Deploy
Staff members
E-business Technical Architect Project Manager
Detailed design document
Implementation staff
Install and configure environments
Implementation staff
Network and Security systems
Graphic designers Usability expert Web developers
Create presentation layer
Product specialists Lead Developer Application developers
Create business logic layer
Product specialists Database developers
Create database layer
Product specialists Integration application developers
Create integration layer
Testing team
Stakeholder testers
Testing of components
- Unit testing
- Link testing
- Integration testing
- Early stakeholder feature testing
Changes
Testing team Stakeholder testers
Deploy live solution
- Performance testing
- User acceptance testing

Once the operational systems and environments have been installed, development of the detailed design commences. This includes parallel development of the functional components in presentation, business logic, data and integration layers, the testing of these components, and their deployment and tuning across operational components.

The functional components developed in the presentation layer are located on web servers, and consist of the web pages that customers see when they connect to the e-business site. Graphic designers create the appearance of this layer through the design of graphical images and the layout of page elements. Web developers then code the site content into this design using technologies such as HTML, DHTML, JavaScript, JSP and ASP, through a range of tools such as Macromedia Dreamweaver and Microsoft Visual Studio. A usability expert is also typically employed to work with the graphic designers and web developers to ensure the presentation layer is simple and easy to use for customers, thus ensuring customer retention. Finally, the presentation layer should be constructed to support all common Internet browsers running on Macintosh and PC platforms to ensure all customers can access the e-business site.

The functional components developed in the business logic layer are used to process customers' requests and apply business-specific rules to appropriate data to produce the required output. These components consist of software code created through custom development or packaged products, and are created by application developers according to the detailed functional design. Once created, components are deployed on application servers according to the operational design. Typically, the business logic layer consists of either open solutions based on J2EE application server products such as WebSphere, iPlanet or BEA WebLogic, or proprietary systems based on C, C++ and COM/DCOM such as many client-server ERP, CRM and financial products. In contrast to open solutions, proprietary – systems typically require an additional integration layer to enable them for customer use over the Internet.

Creating transactional Internet-enabled business logic requires the use of specialist-packaged products or development of custom written code. It is recommended that specialist products that fit requirements be adopted to ensure the solution can be implemented within a short timeframe, thus not delaying the project in the build phase. Additional functionality not included in such products can then be created through customization. Alternatively, if the project requirements cannot be satisfied through deployment of a packaged solution, custom development should be undertaken in the build phase, preferably using open solutions.

The building of the business logic layers requires a database layer for the storage and retrieval of information, such as e-business catalogues and customer and order information. This database layer is located on dedicated hardware servers, and consists of database software installed onto these servers and externally attached storage systems. The database layer is installed and configured by implementation staff members with knowledge of the database product selected in the detailed design. Once these have been installed, specialist database developers work with the application developers to create the design of the database structures to be used by the business logic layer.

Additional systems may be required to participate in the e-business system, such as proprietary products or existing legacy systems. This is achieved through development of an integration layer during the build phase, according to the specifications of the detailed design document. The integration layer is located on separate hardware servers, and consists of integration software products and integration code. This layer is built in parallel with the business logic layer by implementation staff knowledgeable in integration software and integration application developers, and typically requires input from legacy system and product support staff.

Development of design components within each layer is conducted with the participation of testing team members, who work closely with all team members to ensure test plans are written for each component as they are developed. Each test plan documents the component to be tested, the testing conditions in use during the test, the expected input and output for each component, and the results of each testing phase.

As each solution component is completed in the development environment, it is migrated into the testing environment for unit testing to ensure it functions according to the detailed design specifications. The test environment contains duplicated hardware, software, network and security systems from the development environment, and allows for testing to be isolated from development and therefore conducted independently and in parallel.

Following successful completion of unit tests, components are link tested together with related components to ensure correct communication between components. Integration testing is then conducted on the completed set of solution components to determine that the solution satisfies the detailed design requirements. Components are then migrated into the live environment for performance testing to ensure the solution can meet expected levels of demand. This environment consists of the solution components that will be used once the system receives stakeholder approval. If a component fails any stage of the testing process it is returned to the development environment for appropriate changes then retested.

Once the components have passed integration and performance testing, user acceptance testing is conducted in the live environment with a representative group of stakeholders to ensure the solution meets their requirements. Once this is successfully completed, the solution is signed off by stakeholders and becomes ready for deployment into the pilot phase.

It is recommended that user testing by a small group of stakeholders also be conducted during development as this allows the solution to be tuned to more closely meet user requirements. It also allows the project to accommodate any requirements missed during the initial project phases or any subsequent changes in requirements. However, large-scale changes such as switching development methodologies or dramatically altering component functionality, are not recommended as they may seriously impact the successful delivery of the project.

Finally, during the build phase each team member contributes towards the project documentation set. This includes the creation of the test plans and their results, creation of installation and configuration documents for the design components, and a preliminary set of operational management documents detailing how to manage the complete solution. These documents are progressively amended until the handover phase as operational experience is gained with the solution.

1.8 Pilot phase

Once the solution has been built and tested, and passed user acceptance, a pilot is conducted by a representative sample of end users who will utilize the solution in their daily work.

The project manager, technical architect and testing staff assess the ongoing results of this phase, and any modifications that may be required are submitted through the build phase processes to the appropriate staff members. The project staff members also begin training internal support staff in the operation and management of the new solution. This requires input from all project team members, and should be monitored by the project manager and e-business technical architect.

At this stage in the project lifecycle, all functional requirements of the solution should have been delivered. However, solutions frequently require modifications to ensure smooth deployment. These typically involve fixing minor bugs missed during testing, or small changes to the solution to more closely fit requirements. They may also include modifications due to operational issues such as performance, availability and scalability, which may require additional work before the system is put into full live production.

The pilot phase also provides a valuable opportunity to test the transition to the new system for internal staff, and for integrating the new technology with existing systems. Frequently new solutions are swapped in overnight with minimal or no training for staff, with resulting confusion and loss of productivity. Deployment of the solution in a limited pilot allows the business to determine how best to adapt staff work processes to the new system to ensure a smooth transition and to determine training requirements and prepare training materials. It also provides a mechanism to determine how other business technology systems will be affected by the transition to the new system, and allows time to adapt these if required to ensure all technology related issues are solved before implementation.

At this stage in the project lifecycle, major changes to functionality should be avoided, as these will require considerable additional work in new design and build phases, and typically result in considerable delays and greatly increases the chances that the project will not complete successfully.

Such changes can be avoided through management of stakeholder expectations and by ensuring that the correct processes are used throughout the project. This includes involving stakeholders in the initial project planning and requirements gathering phases, by ensuring that the design and auditing phases are properly conducted, and by involving stakeholders in early testing of the solution. Stakeholder management also requires creating awareness of the serious repercussions introduced by requesting major changes in the latter project phases, and by shifting requests for changes or additional functionality into subsequent projects.

1.9 Implementation phase

The implementation phase follows successful deployment and use of the solution in the pilot phase. This phase involves deployment of the solution into 'live' production environments across the company to all appropriate end users and other stakeholders. At this stage in the project lifecycle all requested functionality should be addressed and the stakeholders should have signed off approval for the solution.

Staff members required during the implementation phase typically include infrastructure, implementation and support staff to roll out and tune solution components, development and testing staff to provide additional development and testing services if required, and oversight by the e-business technical architect and project manager. In addition, team members are required to update project documentation throughout the implementation phase.

The implementation phase occurs through a staged deployment of the e-business solution to groups of users, rather than to all users simultaneously. This ensures a manageable transition to the new system without straining available resources and adversely affecting business continuity or affecting existing processes.

This staged deployment in turn requires a set of infrastructure and user deployment processes. Infrastructure deployment processes allow for staged deployment of additional or new hardware and software systems to end users if required, such as new desktop computers and associated systems. User deployment processes are designed to maximize user productivity, and include granting user access to the live production environment, and providing support for users during rollout of the solution, such as training users before changeover to the new system, and providing ongoing assistance once users have migrated.

Finally, during the changeover period the new solution should be monitored for each group of users as it is deployed. This ensures that issues occurring during changeover can be tracked and addressed before subsequent deployment of the solution to additional user groups. This requires infrastructure, development and testing staff be available to optimize performance of the solution as it comes under increased usage, and to correct minor bugs that may occur within the solution.

1.10 Handover phase

Once the solution has been successfully implemented, it is formally handed over to internal staff for ongoing maintenance and support. Project handover involves a process of knowledge transfer, including knowledge of the technical design and implementation, project outcomes, support processes and procedures, and the completed documentation set.

Project handover should include a fixed period of support from critical project members with considerable project input, such as the e-business technical architect and lead developer. This ensures the availability of critical project resources for ongoing knowledge transfer back into the company, and assistance with any technical issues that may arise following implementation.

The final element of project handover requires publishing the complete project documentation set within the company. This provides a reference covering the project history, technical decisions, and solution design, and the deployment and support configurations for internal staff and stakeholders. All documentation and project source code should be supplied on CD-ROM and on the company's Intranet site, allowing ready access for company support staff.

1.11 Project documentation

Following project handover and completion, documentation is required as a knowledge repository for the project, and for the developments of subsequent versions of the e-business system requested by users. In such cases, the documentation provides a detailed understanding of decisions that were made, and the manner in which technologies were implemented, to ensure they do not introduce problems into stable live production systems.

Due to their role in designing and implementing the solution, the e-business technical architect has overall responsibility for the creation and delivery of project technical documentation. Similarly, the non-technical project documentation is the responsibility of the project manager. Non-technical project documentation includes documents concerned with the ongoing management of the complete project by the project manager such as training plans, the project risk register, meeting minutes between project members and company staff, budgets, and the project plan.