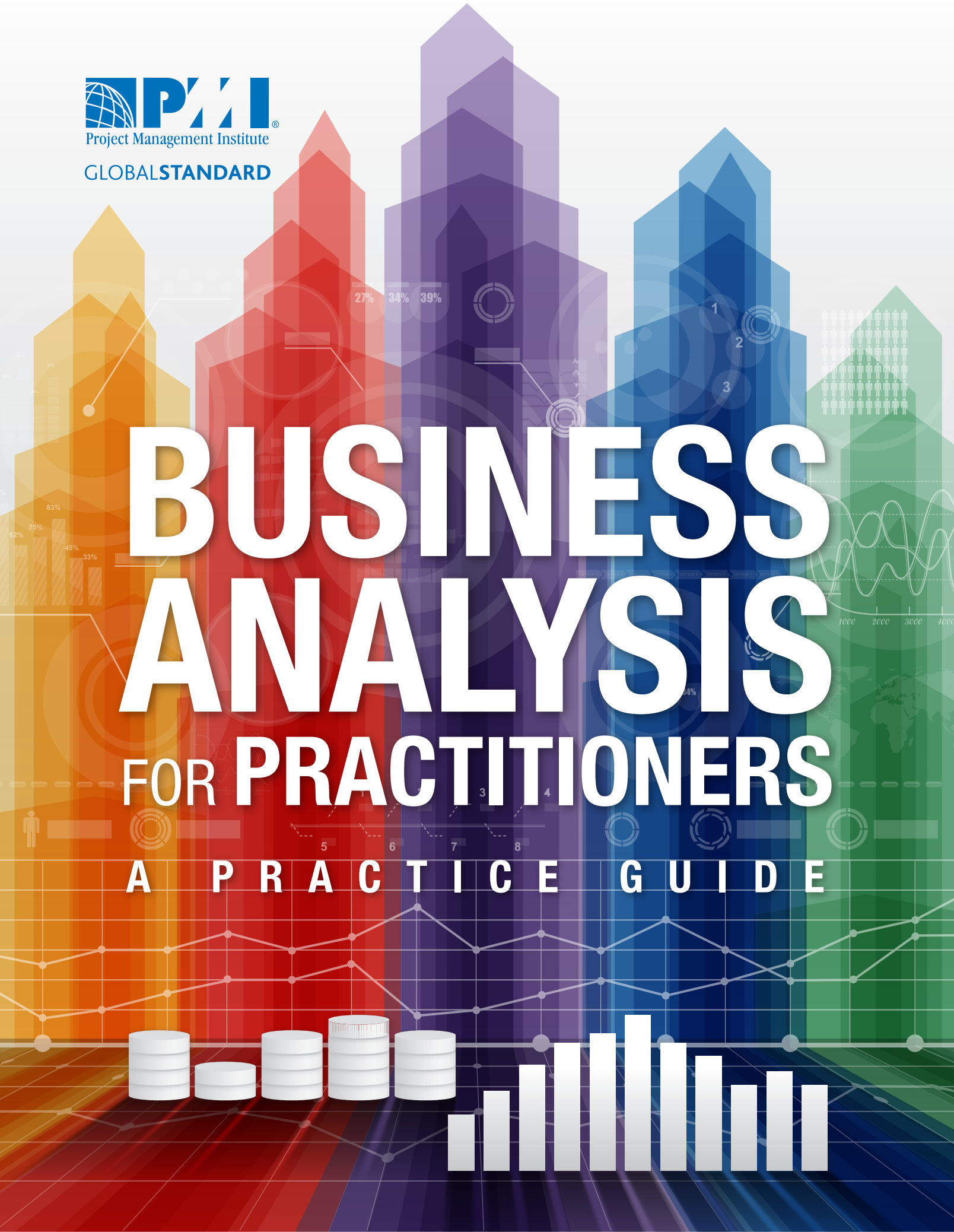
Project Management Institute

GLOBALSTANDARD

BUSINESS ANALYSIS FOR PRACTITIONERS

A PRACTICE GUIDE



Project Management Institute

BUSINESS ANALYSIS FOR PRACTITIONERS: A PRACTICE GUIDE

Library of Congress Cataloging-in-Publication Data

Business analysis for practitioners : a practice guide.
pages cm

Includes bibliographical references.
ISBN 978-1-62825-069-5 (pbk. : alk. paper) -- ISBN 1-62825-069-0 (pbk. : alk. paper) 1. Project management.
2. Business planning. 3. Management. I. Project Management Institute.

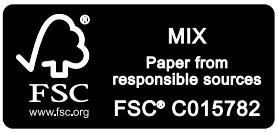
HD69.P75B8745 2015

658.4’013--dc23

ISBN: 978-1-62825-069-5

Published by:
Project Management Institute, Inc.
14 Campus Boulevard
Newtown Square, Pennsylvania 19073-3299 USA
Phone: +610-356-4600
Fax: +610-356-4647
Email: customercare@pmi.org
Internet: www.PMI.org

10 9 8 7 6 5 4 3 2 1



NOTICE

The Project Management Institute, Inc. (PMI) standards and guideline publications, of which the document contained herein is one, are developed through a voluntary consensus standards development process. This process brings together volunteers and/or seeks out the views of persons who have an interest in the topic covered by this publication. While PMI administers the process and establishes rules to promote fairness in the development of consensus, it does not write the document and it does not independently test, evaluate, or verify the accuracy or completeness of any information or the soundness of any judgments contained in its standards and guideline publications.

PMI disclaims liability for any personal injury, property or other damages of any nature whatsoever, whether special, indirect, consequential or compensatory, directly or indirectly resulting from the publication, use of application, or reliance on this document. PMI disclaims and makes no guaranty or warranty, expressed or implied, as to the accuracy or completeness of any information published herein, and disclaims and makes no warranty that the information in this document will fulfill any particular purposes or needs. PMI does not undertake to guarantee the performance of any individual manufacturer or seller's products or services by virtue of this standard or guide.

In publishing and making this document available, PMI is not undertaking to render professional or other services for or on behalf of any person or entity, nor is PMI undertaking to perform any duty owed by any person or entity to someone else. Anyone using this document should rely on his or her own independent judgment or, as appropriate, seek the advice of a competent professional in determining the exercise of reasonable care in any given circumstances. Information and other standards on the topic covered by this publication may be available from other sources, which the user may wish to consult for additional views or information not covered by this publication.

PMI has no power, nor does it undertake to police or enforce compliance with the contents of this document. PMI does not certify, test, or inspect products, designs, or installations for safety or health purposes. Any certification or other statement of compliance with any health or safety-related information in this document shall not be attributable to PMI and is solely the responsibility of the certifier or maker of the statement.

TABLE OF CONTENTS

LIST OF TABLES AND FIGURES

PREFACE

Business Analysis for Practitioners: A Practice Guide is a complementary document to PMI's foundational standards. This practice guide provides guidance on how to apply effective business analysis practices on programs and projects and to drive successful business outcomes. This practice guide provides those with an interest in and commitment to the business analysis discipline the following:

- Diverse collection of both long-established and recent business analysis techniques and practices, defined and explained by experienced business analysis professionals and practitioners; and
- Description of how these techniques and practices can be used including many specific examples.

The information in this practice guide will help readers to:

- Consider which practices and techniques are appropriate for use in their own organizations, and
- Consider how to adapt and adjust techniques and practices to meet organizational and cultural needs without diluting the quality of business analysis which they support.

This practice guide is intended to encourage discussion related to areas of practice where there may not yet be consensus. The discipline of business analysis and its associated roles continue to evolve. Some of the most significant drivers of this evolution are:

- Increased business focus on the ability to accommodate rapid change,
- Increased project focus on delivering value as efficiently as possible, and
- New and evolving approaches for stakeholders and project team members to collaborate with each other to deliver successful projects, which drive business value.

Additionally, the choice of business analysis practices—and how organizations tailor what they choose to implement—is highly dependent on organizational, cultural, and methodological norms. These choices are also impacted by how much change an organization is willing and able to embrace. There is no expectation that every practitioner of business analysis will use every technique noted in the practice guide, for example:

- Some practitioners may consider some of the techniques to be traditional and therefore too confining. PMI recognizes that agile practitioners may desire more adaptive techniques.
- Other practitioners may find that some of the techniques are too new and would potentially introduce risk or complexity.

With all of these considerations in mind, *Business Analysis for Practitioners: A Practice Guide* offers these practices as a starting point to identify thought processes and approaches that may improve how organizations and practitioners approach and achieve effective business analysis.

PMI introduced this practice guide to identify useful approaches for integration with PMI foundational standards. Practice guides are developed by leading experts in the field, and this practice guide is no exception. Practice guides use a relatively new process that provides reliable information while reducing the time required for development and distribution. PMI defines a practice guide as a standards product that provides supporting supplemental information and instructions for the application of PMI standards. Practice guides are not full consensus-based standards and do not go through the exposure draft process. However, the resulting work may be introduced later as a full consensus standard and, if so, will then be subjected to PMI's documented development process for such standards.

1

INTRODUCTION

1.1 Purpose of this Practice Guide

The practice guide describes the work of business analysis and identifies the tasks that are performed in addition to the essential knowledge and skills needed to effectively perform business analysis on programs and projects. This practice guide is applicable to all programs and projects, regardless of whether these are focused on products, services, or process improvement. The concepts and techniques described in this practice guide are implementation-independent and can be used to develop manual or automated solutions, using any type of project life cycle.

The purpose of this practice guide is to define what business analysis is and to demonstrate the practical application of the discipline. This practice guide accomplishes the following:

- Provides a practical discussion of the business analysis work,
- Defines what the work of business analysis is as it relates to programs and projects,
- Discusses why the work is important,
- Provides specific examples of how the work is performed,
- Explains how different types of project life cycles impact the timing and type of business analysis work performed,
- Highlights areas where business analysts should collaborate with other team roles for improved program and project performance, and
- Fully aligns to the tasks, knowledge, and skills that comprise business analysis as identified by the extensive role delineation study conducted for PMI in 2013.

1.2 Need for this Practice Guide

When business analysis is properly accounted for and executed on programs and projects, the following benefits are achieved:

- High-quality requirements are produced resulting in the development of products and services that meet customer expectations;
- Stakeholders are more engaged in the process and buy-in is more readily achieved;
- Projects are more likely to be delivered on time, within scope, and within budget;

- Implemented solutions deliver business value and meet stakeholder needs; and
- Organizations develop competencies in business analysis that are reusable for future projects.

For many organizations, effective business analysis is not an integral part of their project work. As a result, projects are not delivering the intended business value. In 2014, PMI reported the following:[1]

- In the past 12 months, 64% of the completed projects successfully met their original goals and business intent.
- In the past 12 months, 16% of projects that started were deemed failures.
- "Inaccurate requirements gathering" was reported by 37% of organizations as a primary cause of project failure.
- Poor requirements management practices are the second leading cause of project failure, second only to changing organization priorities.

This research clearly shows that organizations continue to experience project issues associated with poor performance of requirements-related activities. Requirements management accounts for a significant portion of the work performed within business analysis. Organizations that have mature business analysis practices in place today are dramatically improving the probability of project success, but those that do not are seeing the costly effects.

PMI has made a commitment to address the project problems identified through this research. This practice guide has been developed to help the industry address the project-related issues associated with requirements and business analysis. Through the development of this practice guide and through the release of other PMI products and services in business analysis, PMI is providing the resources needed to help organizations successfully complete more of their critical initiatives. PMI's business analysis initiatives are based on extensive market research. This research provides a better understanding of how to improve business analysis practices on programs and projects, which will lead to more tangible business outcomes and help organizations exceed customer expectations.

1.3 PMI's Increased Focus on Business Analysis

Requirements have always been a concern in project management. As the focus and importance of requirements work have continued to gain more attention in the industry, PMI standards have continued to evolve to recognize the significance of requirements in programs and projects. *A Guide to the Project Management Body of Knowledge (PMBOK® Guide)* – Fourth Edition[2] was expanded to include the Collect Requirements process within the Project Scope Management Knowledge Area and *A Guide to the Project Management Body of Knowledge (PMBOK® Guide)* –

[1] Hillman, Amy. 2013. The Rise in Business-Analytics Degrees. Retrieved from www.huffingtonpost.com/amy-hillman/the-rise-in-businessanaly_b_3273749.html

[2] Project Management Institute. 2014. *A Guide to the Project Management Body of Knowledge* (*PMBOK® Guide* – Fourth Edition), Newtown Square, PA: Author.

Fifth Edition[3] was expanded to include the Project Stakeholder Management Knowledge Area. PMI is now moving forward on the next evolution of this work by developing this practice guide dedicated to business analysis and, subsequently, may develop a full consensus-based standard. As the global environment becomes more complex, organizations that take a proactive approach to requirements activities will improve their competitive advantage by reducing waste and delivering projects that provide business value.

As organizations begin to recognize how to use business analysis to their competitive advantage, there is an increasing demand for practitioners with the required business analysis skills. According to the U.S. Bureau of Labor Statistics, business analysis jobs are predicted to increase 19% by the year 2022.[4] With the demand for skilled practitioners and the increased emphasis on improving business analysis practices on programs and projects, research indicates that there is a growing need for professionals to acquire the skills needed to fill these critical positions.

1.4 Intended Audience for the Guide

This practice guide is intended for anyone who is responsible for performing business analysis work whether they hold the title of business analyst or not. This practice guide was developed to help practitioners obtain improvements in overall competency levels and in the application of business analysis on programs and projects.

1.5 What is Business Analysis?

Business analysis is the application of knowledge, skills, tools, and techniques to:

- Determine problems and identify business needs;
- Identify and recommend viable solutions for meeting those needs;
- Elicit, document, and manage stakeholder requirements in order to meet business and project objectives;
- Facilitate the successful implementation of the product, service, or end result of the program or project.

In short, business analysis is the set of activities performed to identify business needs and recommend relevant solutions; and to elicit, document, and manage requirements.

This broad definition suggests that business analysis involves effort in a variety of domains: from identifying business needs to solution implementation. Within each of these domains, there are a series of supporting tasks. Each of these tasks are defined and explored within this practice guide. The tasks refine the broad definition and provide specific information about other important aspects of business analysis, such as, facilitating the identification of problems or opportunity analysis for portfolio investment, understanding the business environmental context and constraints, analyzing requirements, verifying requirements, evaluating solutions, etc. Together, the domains and the tasks that are performed within them provide a thorough definition of business analysis.

---

[3] Project Management Institute. 2014. *A Guide to the Project Management Body of Knowledge* (*PMBOK® Guide* – Fifth Edition), Newtown Square, PA: Author.

[4] Business analysts are categorized under management analysts. Refer to report which can be found at http://www.bls.gov/ooh/business-and-financial/management-analysts.htm

Business analysis is conducted in support of many business initiatives, including programs and projects, as well as ongoing operational activities, such as monitoring, modeling, and forecasting. While the primary focus of this practice guide is business analysis in support of programs and projects, the practices herein apply wherever business analysis is conducted.

1.6 Who Performs Business Analysis?

Business analysis may be performed by any individual who is responsible for performing the work regardless of the person's title. In this practice guide, the person(s) who performs business analysis tasks in the context of programs and projects will be referred to as a business analyst. The term is being used in the broad sense and represents all the roles that are responsible for performing the business analysis tasks within their organization and specifically the business analysis tasks on programs and projects.

1.6.1 Skillset and Expertise Needed for the Business Analysis Role

A number of varied skills and competencies are needed in order to perform the business analysis role effectively. As a business analyst becomes more adept at these skills and acquires more project experience, the competency level of the business analyst increases. Many of the interpersonal skills leveraged by project managers are equally important to the practice of business analysis. The following is a partial list of some important skills and expertise for anyone performing business analysis activities on programs and projects:

- Analytical skills,
- Business and industry knowledge,
- Communication skills, including strong business writing and verbal communication skills,
- Conflict management,
- Creative and critical thinking,
- Cultural awareness,
- Decision making,
- Facilitation,
- Familiarity with multiple project and development methodologies,
- Influence,
- Issue management skills,
- Leadership skills,
- Learning skills,
- Negotiation skills,
- Organizational skills,
- Political awareness,

- Presentation skills,
- Problem solving,
- Systems thinking,
- Technical awareness, and
- Ability to work effectively in a team environment, including virtual teams.

1.6.2 How Organizations Implement Business Analysis

This practice guide presents the work of business analysis and does not define the specifics of the business analysis role. The reason for this approach is because the roles are defined in various ways across organizations. Roles are influenced by the type of industry; size of the organization; maturity of the organization in terms of program management, project management, and business analysis practices; and the type of project life cycle in use.

While organizations implement roles in a variety of forms, it is far more effective to define what business analysis is than to specify what comprises the role of the business analyst. An organization may find that business analysis tasks for a project are completed best by assigning a team of business analysts to the work. The work could also be completed by one business analyst, by someone assigned to perform a combined PM/BA (hybrid) role, or other combinations. Ultimately for project success, the important factor is that the business analysis activities are being performed effectively, consistently, and with sufficient quality. It is less important to know the title of the person performing the business analysis work.

Organizations today may find that business analysis is being performed within their organization by one or more of these roles:

- Agile team members;
- Business architects;
- Business intelligence analysts;
- Business process analysts;
- Business subject matter experts;
- Data, functional, operational, systems, or user experience analysts;
- Enterprise business analysts;
- Product managers or product owners;
- Project managers;
- Requirements, software requirements, systems, or value engineers; and
- Requirements managers.

1.6.3 The Relationship Between the Project Manager, Business Analyst, and Other Roles

The project manager and business analyst serve in critical leadership roles on programs and projects. When these roles work in partnership and collaborate effectively together, a project will have a much higher chance of

being successful. Yet the relationship between project managers and business analysts is not always optimally aligned and, consequently a division between the roles performing these activities occurs. Instead of building a close partnership, the roles work independently and at times at odds with one another.

Confusion exists between project managers and business analysts, because there is a perceived overlap of the work that each is responsible for performing. Confusion also exists because there are inconsistent definitions and use of the role across industries, organizations, and departments within the same organization. Confusion continues to build as the role evolves, and organizations that recognize the value of business analysis are beginning to employ more business analysts within their organizations.

This practice guide is intended to clarify these roles through the use of *collaboration points*. These visual callouts are intended to emphasize areas where collaboration between the project manager and business analyst is important and critical to project success. This practice guide also explains the areas of perceived overlap and explains how the work is similar but not the same. Collaboration points are also used to call out opportunities for business analysts to work together with other roles in support of programs and projects.

1.6.4 The Need to Build the Relationships

By providing the industry with a greater understanding of the work performed within business analysis and explaining how it is essential to the overall work of the project, this practice guide is intended to improve the collaboration between these critical roles.

When the project manager and business analyst are not in sync, there are tangible and intangible impacts to project success. When there is a lack of synergy between project managers and business analysts, there are project inefficiencies, critical work is overlooked or duplicated, stakeholders are confused, and the project team fails to operate at an optimum level of efficiency. Taking actionable steps to bridge the gaps between the roles should provide positive impacts to project performance and, ultimately, organizational success.

1.7 Definition of Requirement

In the *PMBOK® Guide* – Fifth Edition, the term requirement is defined as "a condition or capability that is required to be present in a product, service, or result to satisfy a contract or other formally imposed specification."

A requirement represents something that can be met by a product or service, and can address a need of the business, person, or group of people. A requirement should be independent of the design of the solution that addresses it. A requirement may explain a feature that is to be met by a product or software component. When a specific type of requirement is under discussion, the term requirement is preceded by a qualifier such as stakeholder, business, or solution.

1.7.1 Who has the Responsibility for the Requirements?

The responsibility for defining requirements should be assigned to resources that have sufficient business subject matter expertise and decision-making authority. The role with responsibility to conduct business analysis may depend upon the project life cycle, but, in any case, should be assigned to resources with sufficient business

analysis skills and expertise. The project manager is accountable for ensuring that requirements-related work is accounted for in the project management plan and that requirements-related activities are performed on time and within budget and deliver value.

1.7.2 Requirement Types

Requirements are specified for the purpose of clarifying and communicating a business need or required capability. This practice guide uses the term requirement in the broad sense; therefore, when performing the work of requirements elicitation, documentation, and requirements management, it is important to understand the type of requirement being specified. Is the stated requirement a business need, customer need, or a particular stakeholder group need? To provide clarity and context to the issue, requirements are often categorized by type.

In the *PMBOK® Guide* – Fifth Edition, the primary types of requirements discussed include project requirements, product requirements, quality requirements, and stakeholder requirements. Product requirements are the primary focus of this guide and can be further categorized with additional qualifying terms. The following requirement types are discussed in this practice guide and are assumed to be in scope for the elicitation and analysis efforts on projects:

- **Business Requirements.** Describe the higher-level needs of the organization as a whole, such as business issues or opportunities, and reasons why a project has been undertaken.
- **Stakeholder Requirements.** Describe the needs of a stakeholder or stakeholder group, where the term stakeholder is used broadly to reflect the role of anyone with a material interest in the outcome of an initiative, and could include customers, suppliers, and partners, as well as internal business roles.
- **Solution Requirements.** Describe the features, functions, and characteristics of a product, service, or result that will meet the business and stakeholder requirements. Solution requirements are further grouped into functional and nonfunctional requirements.
- **Functional Requirements.** Describe the behaviors of the product.
- **Nonfunctional Requirements.** Describe the environmental conditions or qualities required for the product to be effective.
- **Transition Requirements.** Describe temporary capabilities, such as data conversion and training requirements, and operational changes needed to transition from the current state to the future state.

Two other types of requirements are project requirements and quality requirements. These requirement types are not part of the business analysis effort. These requirements are part of the project work and could be delegated to a business analyst, but are typically the responsibility of the project manager. Since these types are outside the scope of business analysis, they are not discussed in this practice guide.

- **Project requirements** are defined by PMI as “the actions, processes, or other conditions the project needs to meet.” These requirements focus on aspects of project execution.
- **A quality requirement** as defined by the *PMBOK® Guide* – Fifth Edition is “a condition or capability that will be used to assess conformance by validating the acceptability of an attribute for the quality of a result.”

In business analysis, nonfunctional requirements are often referred to as quality of service requirements. Quality of service requirements are not quality requirements. A quality of service requirement describes a quality of the product while a quality requirement describes a quality characteristic of a project deliverable. To avoid any confusion between quality requirements and quality of service requirements, this practice guide uses the term nonfunctional requirements when referring to the category of requirements that describe product quality conditions.

In some organizations, requirements are managed by having separate requirement documents created for each type of requirement; these requirements may also exist in one document separated by document sections. When requirements are managed with a requirements management tool, the requirement type is a characteristic of the requirement that is determined when the requirement is added to the online repository. Regardless of how the types are managed, it is important to ensure that requirement types covered by the project are identified in business analysis planning and properly addressed during elicitation and analysis activities.

1.8 The Structure of the Practice Guide

This practice guide organizes the work of business analysis into five domains. These domains were defined originally as part of the conceptual framework identified through a role delineation study completed for PMI in 2013. The five domains of business analysis practice as identified by the role delineation study are: Domain 1–needs assessment, Domain 2–planning, Domain 3–analysis, Domain 4–traceability and monitoring, and Domain 5–evaluation.

These domains are reflected in Sections 2 through 6 of this practice guide. To minimize confusion between the planning that occurs in project management and the planning that occurs in business analysis, Section 3 in this practice guide is titled Business Analysis Planning. Section 4 is titled Requirements Elicitation and Analysis to more appropriately reflect the work being performed in the domain, which includes the requirements elicitation tasks as well as the requirements analysis tasks.

1.8.1 Section 2 on Needs Assessment

Section 2 discusses the business analysis work that is conducted to analyze a current business problem or opportunity and to assess the current internal and external environments of the organization for the purpose of understanding what needs to occur in order to attain the desired future state. Some of this work may be undertaken by business analysts before a project is proposed. Section 2 further explains the business analysis tasks to understand the goals and objectives of the organization, define problems and opportunities, assess the current capabilities of an organization, define the desired future state, identify capability gaps, and contribute to the development of a business case for the purposes of proposing viable options that will enable the organization to meet its business objectives. This section presents various techniques for analyzing and assessing the organization as well as valuation techniques for assessing the viability of solution options.

1.8.2 Section 3 on Business Analysis Planning

Section 3 discusses the work that is conducted in order to define the business analysis approach and plan for the completion of the requirements-related activities necessary to meet the needs of the project. Section 3

discusses the use of stakeholder analysis to complete a thorough assessment of the stakeholders who will be participating in the business analysis efforts and discusses all of the process decisions and planning activities that are recommended for constructing an optimal business analysis plan for the project. Section 3 discusses how the selected project life cycle influences the timing and the approach to business analysis planning and describes how the approach will be different based on the life cycle chosen.

1.8.3 Section 4 on Requirements Elicitation and Analysis

Section 4 discusses the iterative nature of the work performed to plan, prepare, and conduct requirements elicitation and to analyze and document the results of that work. A number of elicitation and analysis techniques are defined and explained. Examples are included to provide context and describe how to practically apply these techniques on projects. Different forms of requirement documentation choices are discussed and guidelines for writing high-quality requirements are provided. A large percentage of project time is spent on elicitation and analysis; therefore, this section provides a thorough explanation of concepts in order to help practitioners better perform in these areas.

1.8.4 Section 5 on Traceability and Monitoring

Section 5 covers the comprehensive set of activities for approving requirements and managing changes to requirements throughout the project life cycle. The benefits associated with capturing requirement attributes and building a traceability matrix for a project are discussed. Section 5 provides a formal requirements change process and discusses how changes to requirements are analyzed, assessed, approved, communicated, and managed throughout a project. Baselining requirements, impact analysis, configuration management, and version control are also addressed. Additionally, considerations for a streamlined approach to traceability and monitoring are noted.

1.8.5 Section 6 on Solution Evaluation

Section 6 discusses the business analysis tasks that are performed to validate a solution that is either implemented or ready to be implemented. This section focuses on both qualitative and quantitative evaluation methods; discusses how evaluation criteria and acceptance levels are used to perform an evaluation of the solution; and discusses work performed to evaluate, analyze, and report on the evaluation results. A number of evaluation techniques are defined and examples are shared to demonstrate the practical application of their use.

2

NEEDS ASSESSMENT

2.1 Overview of this Section

A needs assessment consists of the business analysis work that is conducted in order to analyze a current business problem or opportunity. It is used to assess the current internal and external environments and current capabilities of the organization in order to determine the viable solution options that, when pursued, would help the organization meet the desired future state.

This section of the practice guide offers a comprehensive approach for assessing business needs and identifying high-level solutions to address them. It provides ways to think about, learn about, discover, and articulate business problems and opportunities. Thinking through business problems and opportunities with stakeholders is important for all programs and projects; the degree to which a needs assessment is formally documented depends upon organizational and, possibly, regulatory constraints.

2.2 Why Perform Needs Assessments

In business analysis, needs assessments are performed to examine the business environment and address either a current business problem or opportunity. A needs assessment may be formally requested by a business stakeholder, mandated by an internal methodology, or recommended by a business analyst prior to initiating a program or project. As used in this practice guide, a project is a temporary endeavor undertaken to create a unique product, service, or result. A program is a group of related projects, subprograms, and program activities managed in a coordinated way to obtain benefits not available from managing them individually.

Needs assessment work is undertaken before program or project work begins, therefore it is said to involve the preproject activities. However, during the course of a project, should external factors change (e.g., corporate merger, large percentage loss of market share etc.), which influence or impact the project in process, the business analyst will need to revisit the needs assessment and decisions made previously to ensure they are still valid for the situation the business is addressing.

A needs assessment involves the completion of a gap analysis—a technique described in Section 2.4.7—that is used to analyze and compare the actual performance of the organization against the expected or desired performance. Much of the analysis completed during the needs assessment is then used for the development of a business case. It is the needs assessment and business case that build the foundation for determining the project objectives and serve as inputs to a project charter.

When a needs assessment is bypassed, there is often insufficient analysis to adequately understand the business need. The business analyst conducts a needs assessment to help the organization understand a business problem or opportunity in greater detail to ensure that the right problem is being solved. When a formal needs assessment is sidestepped, the resulting solution often fails to address the underlying business problem or fails to solve the problem completely; it is also possible to provide a solution that is not needed or one that contains unnecessary features.

2.3 Identify Problem or Opportunity

Part of the work performed within needs assessment is to identify the problem being solved or the opportunity that needs to be addressed. To avoid focusing on the solution too soon, emphasis is placed on understanding the current environment and analyzing the information uncovered. The business analyst needs to ask "what problem are we solving?" or "what problems do our customers have that this opportunity will address?" The business analyst begins to elicit information to uncover the data necessary to fully identify the problem or opportunity.

2.3.1 Identify Stakeholders

Stakeholder identification is conducted as part of the needs assessment to assess which stakeholders are impacted by the area under analysis. A stakeholder is an individual, group, or organization that may affect, be affected by, or perceive itself to be affected by a decision, activity, or outcome of a program or project.

For example, when an organization wants to take advantage of a new technology or plans to automate current manual processes, stakeholder identification is performed to identify who will be impacted by the changes. There are various ways to locate stakeholders. Some possible methods are described in Section 3 on Business Analysis Planning, where this topic is presented in more depth.

During needs analysis, it is helpful to identify the following stakeholders:

- Sponsor who is initiating and responsible for the project,
- Stakeholders who will benefit from an improved program or project,
- Stakeholders who will articulate and support the financial or other benefits of a solution,
- Stakeholders who will use the solution,
- Stakeholders whose role and/or activities performed may change as a result of the solution,
- Stakeholders who may regulate or otherwise constrain part or all of a potential solution,
- Stakeholders who will implement the solution, and
- Stakeholders who will support the solution.

The affected stakeholders for a needs assessment can be categorized into one of four categories using a responsibility assignment matrix such as a RACI model:

- **R—Responsible.** Person performing the needs assessment,
- **A—Accountable.** Person(s) who approves the needs assessment, including the business case, when warranted,
- **C—Consult.** Person or group to be consulted for input to understand the current problem or opportunity, and
- **I—Inform.** Person or group who will receive the results of the needs assessment.

Example—Consider an insurance company that is interested in reducing the processing times and costs for automobile and homeowner claims. Initially, the organization understands the solution may impact a number of stakeholders across the company. To better understand who to involve in the needs assessment phase, the business analyst develops a RACI matrix to determine the roles and levels of responsibility.

Table 2-1 shows an example of a partial stakeholder list for the insurance company. Other possible stakeholders might be the claims operations manager, claims examiners, partners, and suppliers.

Table 2-1. Example RACI for Assessing Business Need

	Sponsor	Product Manager	Business Analyst	Product Development Team	Mobile Technical Team	Project Manager
Identify problem or opportunity	A	C	R	C	C	
Assess current state of the organization	A	I	R	C	C	
Recommend action	I	A	R	C	C	C
Prepare business case	I	A	R	C	I	I

Collaboration Point—Both project managers and business analysts have an interest in stakeholder identification and RACI analysis. While the project manager is concerned about analyzing the roles across the project, business analysts may perform their analysis to a lower level of detail or may focus on one specific area, such as a needs assessment or requirements elicitation. Each may lend support to the other and work together to perform this work. It is important to ensure efforts are not duplicated.

2.3.2 Investigate the Problem or Opportunity

The business analyst focuses on learning enough about the problem or opportunity to adequately understand the situation, but avoids conducting a complete requirements analysis at this stage. As used here, situation is a neutral word to describe the context about the problem or the opportunity being investigated.

Initially, the business analyst may conduct interviews with stakeholders to investigate the situation and learn about the current environment. The business analyst may also review any existing documentation about current processes, methods, or systems that support the business unit. Process modeling is one technique used to document current "as is" processes of the business. The business analyst may monitor or observe the business performing their work in order to discover elements of the current "as-is" process. This technique is referred to as observation. For more information on observation, refer to Section 4 on Requirements Elicitation and Analysis.

2.3.3 Gather Relevant Data to Evaluate the Situation

Once a broad understanding of the situation is obtained, it is necessary to gather relevant data to understand the magnitude of the problem or opportunity (also known as "sizing up" the situation). The lack of data can result in proposing solutions that are either too small or too large compared to the problem at hand. In other words, the business analyst should attempt to measure the size of the problem or opportunity to help determine an appropriately sized solution.

When no internal data exists or when it cannot be feasibly collected, benchmarking may be performed. Benchmarking is a comparison of the metrics or processes from one organization against a similar organization in the industry that is reporting or finding similar industry averages. Data may not be readily available since competitors will guard nonpublic data closely. Benchmarking may also involve comparing internal organization units or processes against each other. Examples of data suitable for benchmarking include, but are not limited to the following:

- Cycle times for a business process to complete transaction volumes, occurrences of exceptions or problems, and delays caused by the exceptions or problems;
- Amount of money lost per transaction, per sale, by losing a customer, from costs to acquire a new customer, through waste, and from calls to a help desk;
- Website visitors, website conversions, sales inquiries, new accounts, and new policies;
- Potential increases in sales, market share, customer base, and new contracts;
- Market size, potential new market share, current competition, and pricing structures in place; and
- Competitive analysis of products offered, feature and benefit comparisons, pricing, and policies regarding the foregoing.

Once the desired data is assembled, techniques such as Pareto analysis and trend analysis can be used to analyze and structure the data.

2.3.4 Draft the Situation Statement

Once the problem is understood, the business analyst should draft a situation statement by documenting the current problem that needs to be solved or the opportunity to be explored. Drafting a situation statement is not time-consuming, but it is a very important step to ensure a solid understanding of the problem or opportunity the organization plans to address. If the situation statement is unknown, or wrong, or if the stakeholders have a different idea of the situation, there is a risk that the wrong solution will be identified.

The format of a situation statement is as follows:

- Problem (or opportunity) of "a"
- Has the effect of "b"
- With the impact of "c"

Example—Consider the insurance company example that was presented previously. Like many companies, this insurance carrier took advantage of new technologies across its business and has an extensive Internet presence. The organization now wants to exploit mobile technology to improve its claims processing.

Before the development of a project charter, the business analyst drafts a situation statement for inclusion in a formal business case. Using the knowledge gained through the initial needs assessment work, the business analyst considers the identified business need and any initial assessment of the impact that the problem is having on the business. With this background information, the business analyst can then draft the situation statement, such as:

"The cost for processing claims has been rising steadily, increasing at an average rate of 7% per year, over the last 3 years. The existing method for submitting claims either by phone or the Internet involves significant processing delays and has resulted in the need to increase staffing to process the calls and personally investigate the claims."

The problem, as assessed, was one of increased costs to process claims, including additional labor costs.

Note: The financial impacts identified in the situation statement will later be referenced during the completion of the cost-benefit analysis.

2.3.5 Obtain Stakeholder Approval for the Situation Statement

Once the situation statement is drafted, agreement is obtained from the affected stakeholders that were previously identified. This is a key step because the situation statement guides subsequent work for assessing the business need. When a formal situation statement and its approval are skipped, it is difficult to determine whether the essence of the current situation has been captured. Business stakeholders play an important role to ensure that the situation statement correctly defines the situation. Failing to refine the situation statement with the business may result in a solution that addresses only part of the business need or fails to meet the business need at all.

The business analyst initiates and facilitates the approval process, which may be formal or informal, depending on the preferences of the organization. Approval may not occur upon the first review of the situation statement, and there may be a need to revise or reword the statement so that the stakeholders are in agreement with it. The business analyst leverages skills such as facilitation, negotiation, and decision making to lead stakeholders through this process.

2.4 Assess Current State of the Organization

Once the relevant stakeholders agree on the problem that needs to be solved or the opportunity the organization wishes to exploit, the situation is analyzed in greater detail to discover important components such as the root causes of the problems identified in the situation statement. As stated previously, the assessment is not a complete

current state analysis. It is intended to understand current organizational goals and objectives, root causes of problems that prevent achievement of these, goals and/or any important contributors to opportunities that could help attain them.

2.4.1 Assess Organizational Goals and Objectives

Depending on the organization, business strategy documents and plans may be available for review by the business analyst in order to acquire an understanding of the industry and its markets, the competition, products and services currently available, potential new products or services, and other factors used in developing organizational strategies. In the absence of such plans, it may be necessary to interview stakeholders to determine this information.

The organizational goals and objectives are an important input used by the business analyst when they begin documenting the business requirements. Goals and objectives that are relevant to the situation provide the context and direction for any change or solution that addresses the business need.

Business requirements are goals, objectives, and higher-level needs of the organization that provide the rationale for why a project is being undertaken. Business requirements are defined before a solution is determined and recognize what is critical to the business and why. For additional information pertaining to business requirements, see Section 4 on Requirements Elicitation and Analysis.

2.4.1.1 Goals and Objectives

Organizational goals and objectives are often revealed in internal corporate strategy documents and business plans. Corporate strategies translate goals identified in business plans into actionable plans and objectives. Goals are typically broad-based and may span one or more years. Objectives, on the other hand, are used to enable goals; these are more specific and tend to be of shorter term than goals, often with durations of 1 year or less.

An organization accomplishes its objectives in various ways, including through programs and projects. The most common link between goals and objectives and programs and projects is the business case. Figure 2-1 shows an example of the hierarchical relationship between goals, where goals have one or more objectives and may have any number of business cases and various tactical plans to support them. The approved business cases are used as inputs to programs and projects. However, not all projects have a business case associated with them, but this section will focus on those that do.

Goal and objective modeling approaches are discussed in Section 4.10.7 on Scope Models.

2.4.1.2 SMART Goals and Objectives

If goals and objectives are not specified or are not clear, the business analyst should document them to establish the basis for subsequent work that relies on them. Well-written goals and objectives are also said to be “SMART” as summarized in Figure 2-2. Note there are subtle variations for writing SMART goals and/or objectives; this example is one of the more common approaches used today.

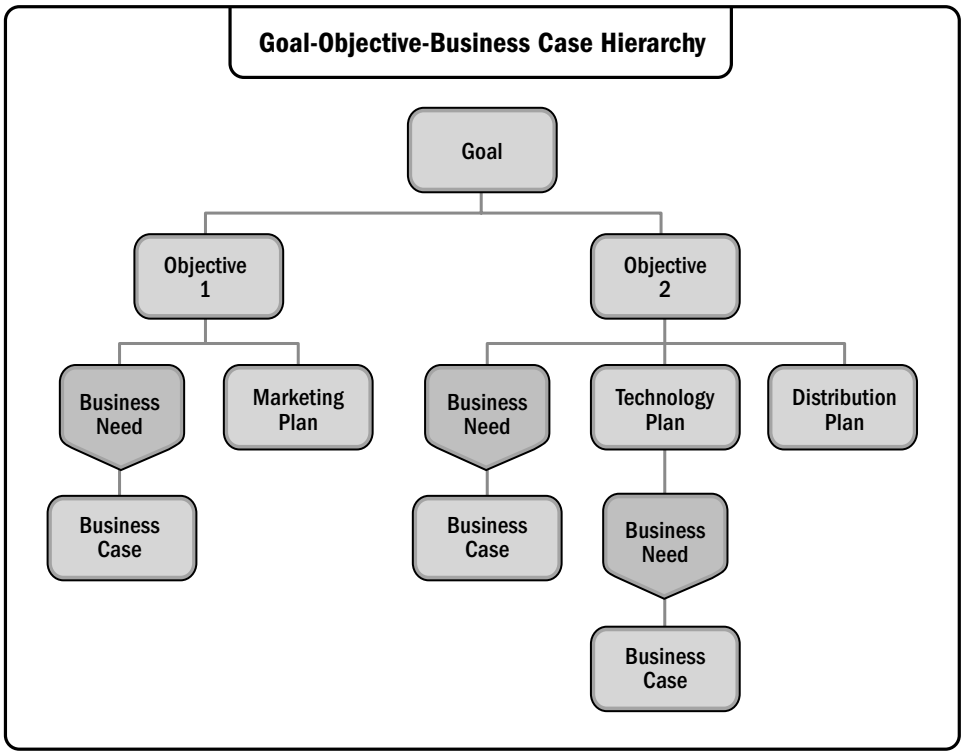


Figure 2-1. Example Hierarchy from Goals to Business Cases

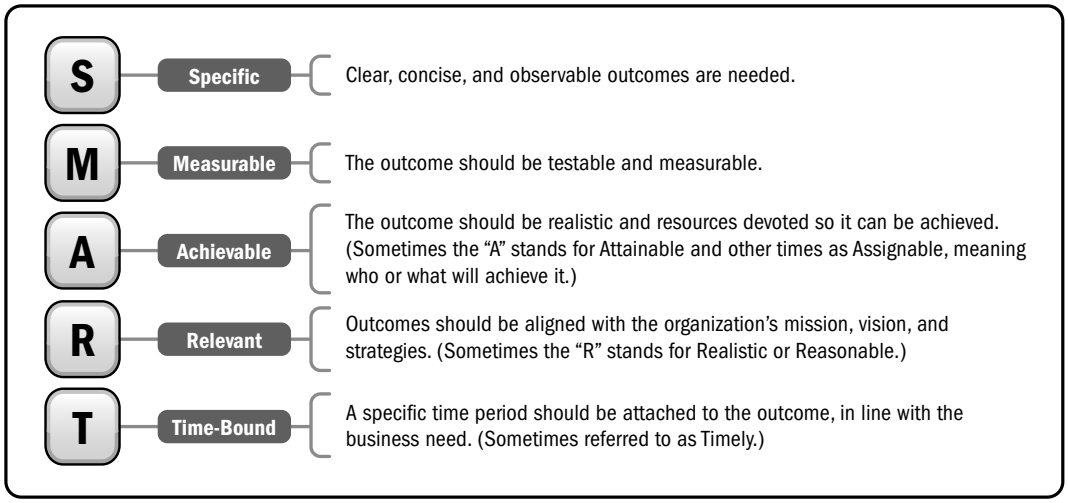


Figure 2-2. SMART Goals and Objectives

Example—In the previous insurance company example, the insurance company had a goal of reaching $5 billion in revenue within 5 years, with a 20% net profit margin. The company also had the following supporting objectives for the coming year:

- Increase revenues by 10% by December 31 (necessary to help them reach their 5-year goal),
- Decrease overall claims costs by 5% in the same time period,
- Reduce time needed to process claims by 6.25% in each quarter of the year.

In summary, goals and objectives are important to needs assessment, because they provide the context and provide direction for any change that addresses the business need. Ideally, except for unforeseen problems and opportunities, all programs and projects directly support the stated business goals and objectives. Programs and projects are linked to the goal and objectives through the business case. Business cases are assembled as one of the final tasks in the needs assessment and are discussed in further detail later on in this section. Even without a formal business case, goals and objectives should be leveraged to guide the direction of business analysis.

2.4.2 SWOT Analysis

In the absence of formal plans, the business analyst may use SWOT analysis to help assess organizational strategy, goals, and objectives. SWOT (standing for strengths, weaknesses, opportunities, and threats) is a common method used to facilitate discussions with stakeholders when articulating high-level and important aspects of an organization, especially as it pertains to a specific situation.

SWOT uses the four categories mentioned previously and provides an additional context for analyzing the business need. It helps to translate organizational strategy into business needs. SWOT investigates the situation internally and externally as follows:

- Internally:
  - Shows where the organization has current strengths to help solve a problem or take advantage of an opportunity. Examples of strengths include a knowledgeable research staff, strong brand reputation, and large market share.
  - Reveals or acknowledges weaknesses that need to be alleviated to address a situation. Weaknesses may include low recognition in the market, low capitalization or tax base, and bad publicity due to real or imagined failures.
- Externally:
  - Generates potential opportunities in the external environment to mitigate a problem or seize an opportunity. Examples of opportunities include underserved markets, termination of a competitor's product line, and discovery of a customer need that the organization can satisfy with a new product.
  - Shows threats in the market or external environment that could impede success in solving business needs. Threats may include increased market share by the competition, new products offered by competitors, mergers and acquisitions that increase a competitor's size and clout, and new regulations with potential penalties for noncompliance.

SWOT is a widely used tool to help understand high-level views surrounding a business need. The business analyst may use SWOT to create a structured framework for breaking down a situation into its root causes or contributors.

Example—Figure 2-3 shows a sample SWOT diagram for the insurance company example.

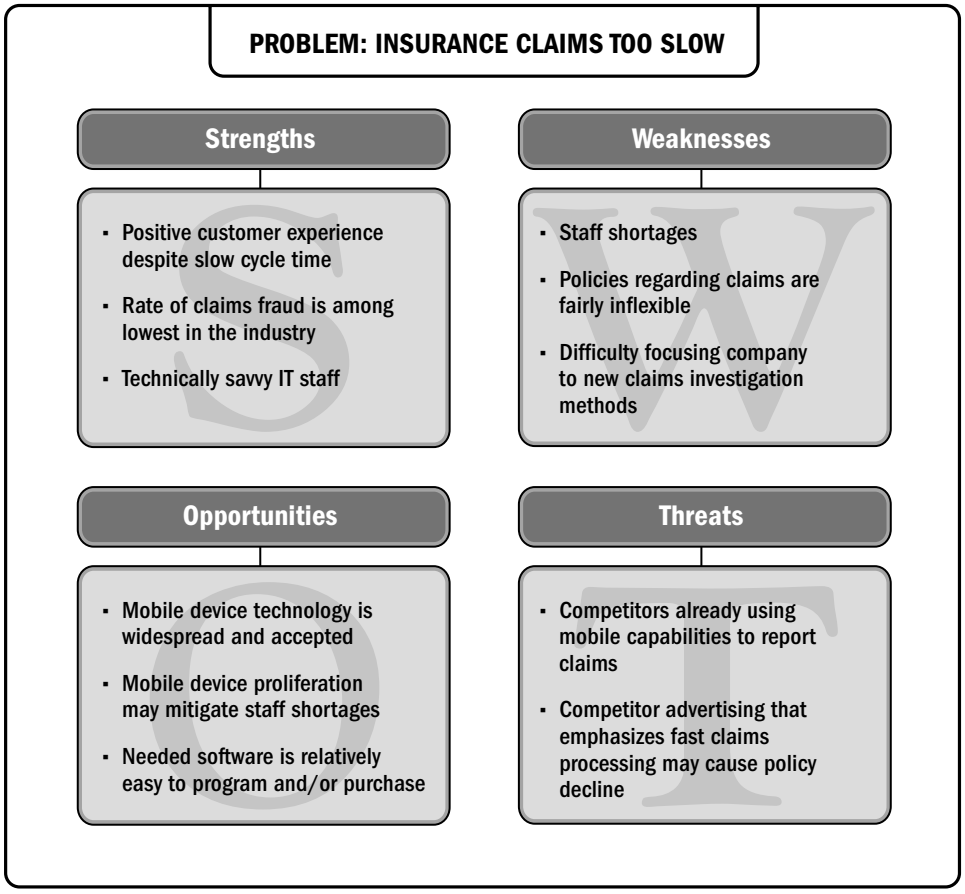


Figure 2-3. Example SWOT for Business Problem

2.4.3 Relevant Criteria

Goals and objectives provide criteria that may be used when making decisions regarding which programs or projects are best pursued. When goals and objectives involve revenue generation, then programs or projects to expand markets or add new products are key. When one of the major objectives is to decrease costs, then programs or projects for process improvement or cost elimination are important.

> *Example*—One objective of the insurance company is to increase revenues 10% by December 31. In this case, revenue is a highly relevant criterion. The organization needs to sell new insurance policies and possibly expand its markets; therefore, programs or projects that improve sales will be significant. Initiatives that support expense reduction are very important too because the company also needs to reduce expenses. Criteria that were identified when reviewing the organizational goals and objectives will be useful later when comparing and rank-ordering potential solution options.

2.4.4 Perform Root Cause Analysis on the Situation

Once a situation is discovered, documented, and agreed upon, it needs to be analyzed before being acted upon. After agreeing on the problem to be solved, the business analyst needs to break it down into its root causes or opportunity contributors so as to adequately recommend a viable and appropriate solution. For purposes of brevity, this practice guide will treat root cause analysis and opportunity analysis as one topic. To clarify these terms, each is defined as follows:

- **Root Cause Analysis.** An analytical technique used to determine the basic underlying reason that causes a variance, defect, or risk. When applied to business problems, root cause analysis is used to discover the underlying cause of a problem so that solutions can be devised to reduce or eliminate them.
- **Opportunity Analysis.** A study of the major facets of a potential opportunity to determine the viability of successfully launching a new product or service to enable its achievement. Opportunity analysis may require additional work to study the potential markets.

There are a number of techniques used to perform root cause and opportunity analysis, and most work for both. This practice guide presents the following common methods:

2.4.4.1 Five Whys

The objective of Five Whys is to ask for the cause of a problem up to five times or five levels deep to truly understand it. A business analyst does not always need to literally ask "why" up to five times. Instead, the Five Whys are used to begin with a problem and ask why it occurs until the root cause becomes clearer. Quite often, business people bring solutions to the project team, but it is essential to first clarify the business problem with a technique like Five Whys before considering solutions. Other techniques may be needed to refine the root cause, but Five Whys is a good starting point.

It is important to ask "why" using appropriate questions and to limit the actual use of the word "why," because it can cause the interviewee to become defensive.

Example—In the insurance company example, a Five-Whys dialog might proceed as shown in Table 2-2.

2.4.4.2 Cause-and-Effect Diagrams

Cause-and-effect diagrams decompose a problem or opportunity to help trace an undesirable effect back to its root cause. These diagrams help to break down the business problem or opportunity into components to aid understanding and generally provide the main aspects of the problem to analyze. They are typically high-level views of why a problem is occurring or, in the case of an opportunity, these views represent the main drivers for why that opportunity exists. Cause-and-effect diagrams are designed to understand the cause of a problem so as not be distracted by its symptoms. These diagrams take a systems view by treating the environment surrounding the problems as the system and by avoiding analysis of the problems imposed by people or staff.

There are several types of cause-and-effect diagrams that could be used to uncover root causes. Most of these techniques can be used along with the Five Whys to dissect a problem. From a practical standpoint, it may be

Table 2-2. Sample Conversation Using Five-Whys Dialog

Role	Question/Reply
Sponsor	We would like to add the ability for policyholders to submit claims from their mobile phones. We figure it would speed up claims processing considerably.
Business analyst	I'm new on this team. Can you help me to understand why this is a problem? ***[Why 1]***
Sponsor	Well, the problem is that claims take too long to process. With a mobile application, we can encourage customers to file claims as soon as an accident or storm happens. Plus, there are other features of smart phones we can exploit, like using their cameras and video technology.
Business analyst	What do you think is the major delay in processing claims? ***[Why 2]***
Sponsor	Partly it's the lag between the time of an incident and when the policyholder files a claim, which can add several days to a week to the process time. The delay also results from our corporate policy that we need to investigate every claim we think will exceed certain limits. That tends to be 80% of all claims.
Business analyst	Can you tell me the reason behind the need to investigate so many claims personally? ***[Why 3]***
Sponsor	We're a pretty conservative company, and to avoid fraud, we like to personally view the damage.
Business analyst	What other alternatives for speeding up claims have you tried in the past, and why didn't they work? ***[Why 4]***
Sponsor	Well, we tried skipping the investigation for all but the highest claim amounts, and our losses jumped way up. We also tried encouraging customers to call us on a dedicated line from their mobile phones. But for some reason they didn't seem to have our number handy or who knows completely why, but we didn't get enough calls to warrant continuing.
Business analysis analyst	What did you attribute the higher losses to? ***[Why 5]***
Sponsor	We found out that many of the damages were not as bad as the claims indicated. I think we overpaid by around 20% if I remember correctly.

sufficient to go two to three levels deep in order to understand the needed elements of root cause. Two of the most useful cause-and-effect diagrams are described as follows:

- **Fishbone Diagrams.** Formally called Ishikawa diagrams, these diagrams are snapshots of the current situation and high-level causes of why a problem is occurring. These diagrams are often a good starting point for analyzing root cause. Fishbone diagrams lend guidance to the causes that will provide the most fruitful follow-up. For example, they often uncover areas in which data is lacking and would be beneficial to collect. However, this technique is not sufficient for understanding all root causes.

  Note: Fishbone diagrams traditionally use the word "effect" because they are cause-and-effect diagrams. However, it could easily be a situation or problem instead.

  There are various ways of depicting a fishbone diagram. Figure 2-4 shows one such rendition. Here are some guidelines for using fishbone diagrams:

  - The problem to be solved is placed at the head of the fish, which can be either facing to the right or left.
  - Use between three to eight causes or categories as an optimum number of causes associated with the problem. Standard categories, when used, can help guide the process and include machines, methods/systems, materials, measurements, people/customers, policies, processes/procedures, and places/locations.
  - For each cause identified, ask why that cause is occurring and label any subcauses discovered. Generally, two to three levels will be sufficient to gain insights into the problem to understand its root causes.

- Look for patterns of causes, which are useful items to measure, model, or otherwise study in more detail. They are more likely to lead to the root cause of the situation. Circle those significant factors as shown in the example in Figure 2-4.

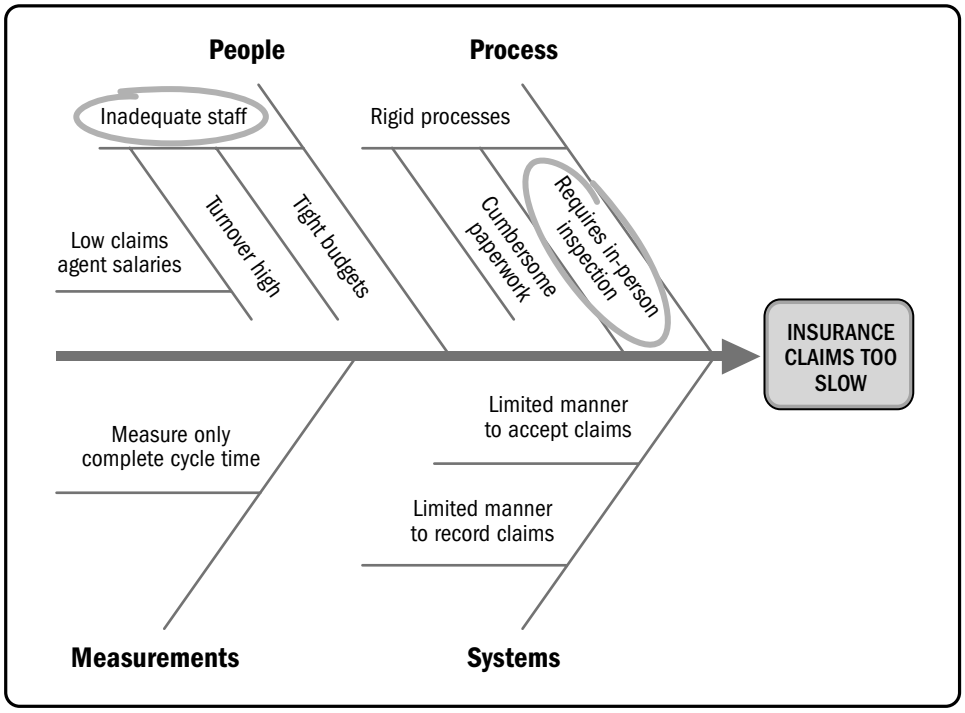


Figure 2-4. Fishbone Diagram Example

- **Interrelationship Diagrams.** This special type of cause-and-effect diagram is helpful for visualizing complex problems that have seemingly unwieldy relationships among multiple variables. These diagrams are most useful for identifying variables, but similar to the fishbone diagram, this technique is not sufficient for understanding all root causes. In some cases, a cause of one problem may be the effect of another. The interrelationship diagram can help stakeholders understand the relationships between causes and effects and can identify which causes are the primary ones producing the problem.

  Constructing an interrelationship diagram helps participants isolate each dimension of a problem individually without it being a strict linear process. Focusing on the individual dimension allows participants to concentrate on and analyze manageable pieces of a situation. When the analysis is complete, the diagram sheds considerable light on the problem, but only after the entire diagram has been assembled.

  Here are some guidelines for using interrelationship diagrams:

  - Identify the potential causes and effects, up to a maximum of ten to be practical.
  - Draw lines between the effects and their causes. The arrows represent the direction of the cause. The arrow starts from the cause factor and points to the effect. See Figure 2-5 and note the factor "budget cuts" and the causing factor "shortage of staff."

- When two factors influence each other, understand which of the two is stronger and note which one. Interrelationship diagrams are most effective when arrows are limited to one direction, namely the stronger influence.
- Factors with the largest numbers of "incoming" arrows are those that are key outcomes of other factors, that is, effects. These often provide good measures of success or can be good items to quantify and monitor. In Figure 2-5, "scheduling delays" is shown to be the most significant effect.
- Factors that have a large number of "outgoing" arrows are the sources of the problem, that is, causes. Label the factor numbers in and out. In Figure 2-5, "budget cuts" is shown to be the most significant cause. Consider prioritizing needs assessments to address the strongest causes first.
- The significant cause and effect factors can be depicted in bold to emphasize them as in the example in Figure 2-5.

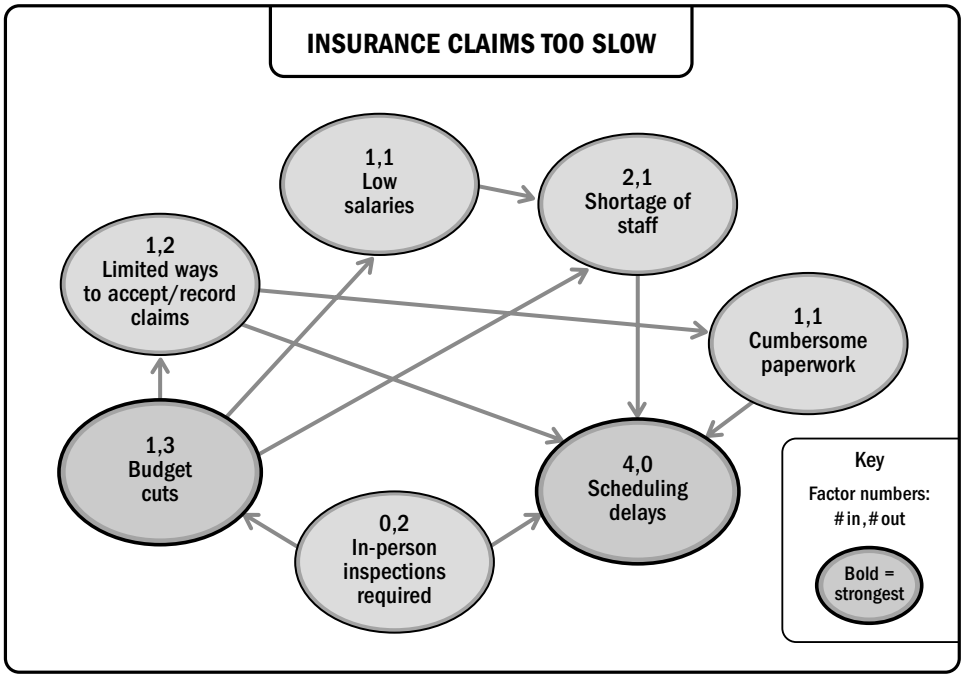


Figure 2-5. Interrelationship Diagram Example

- **Process Flows.** Process flows are used for many things, primarily to document and analyze current and future (e.g., proposed) processes. Another use for these models is to analyze a process for the ways in which a current process contributes to a given problem. In this sense, a process flow can be used as a root-cause analysis tool. A nonoptimal process existing in the current as-is environment may have gaps or missing steps, duplication, or unnecessary and non-value-added steps. Process flows are not useful when analyzing opportunities.

  Process flow modeling can be used in conjunction with root cause analysis as it focuses on how aspects of current processes may contribute to the problem at hand. In the insurance company example, the

scheduling limitations for the in-person inspections are a major contributor to the current situation. Figure 2-6 is an example of a process flow that is used for root cause analysis. Some guidelines for using process flows for problem analysis include:

- Select the situation and place it at the top of the process map to emphasize the problem to be solved. Here it will remain highlighted.
- Identify the steps in the process that result in the problem. Try and keep the number of steps at a high level, often around ten or fewer steps.
- For each step, ask how that step may be contributing to the identified problem. For each cause, probe to find any subcauses and add them in a similar fashion to a fishbone diagram. Figure 2-6 shows a cause of "available times are hard to find," with a subcause of "limited adjusters," and a root cause of "budget cuts."
- Look for steps that have several issues as the best places to find changes or improvements, or perform further investigations to verify and/or measure. Circle the factors that are felt to be significant causes and subcauses of the problem being assessed. Figure 2-6 shows one repeating factor that was noteworthy.

Process flows are described in more depth in Section 4 on Requirements Elicitation and Analysis.

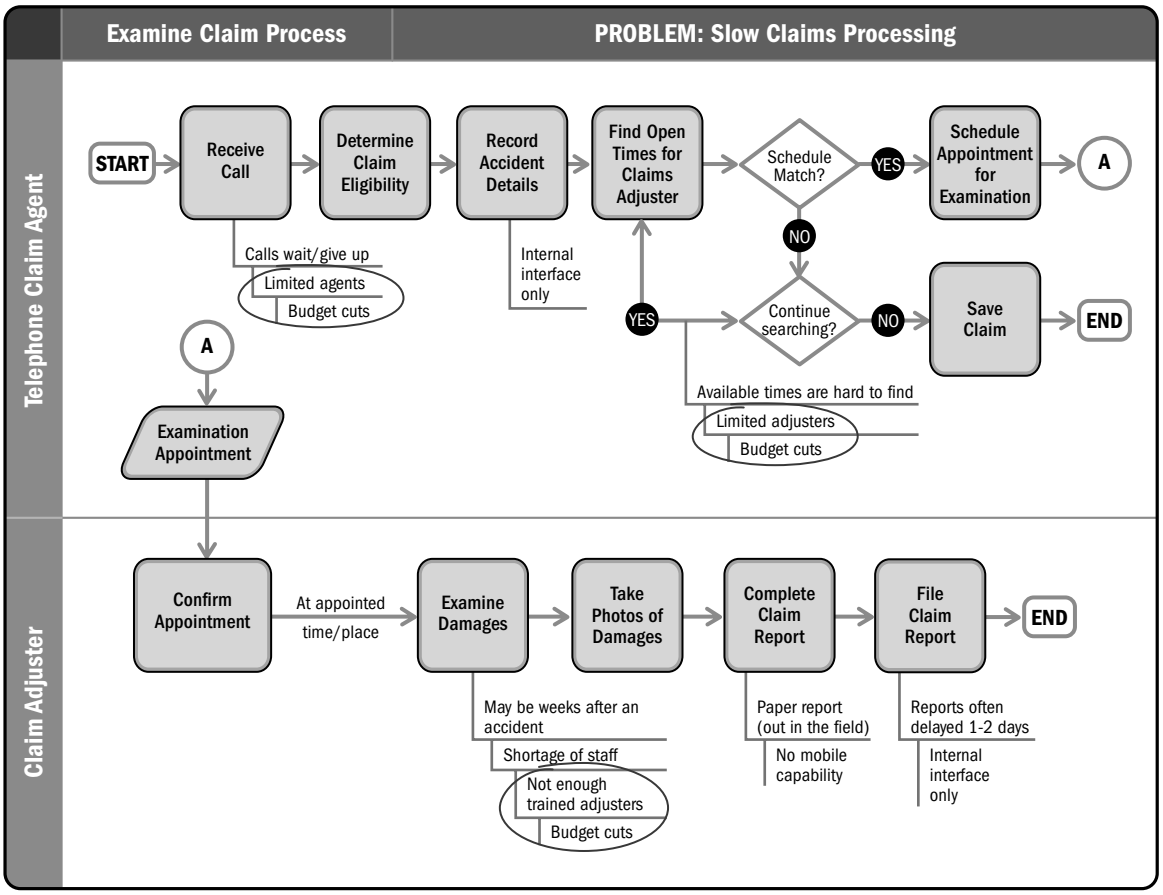


Figure 2-6. Process Flow with Root Cause Analysis Example

2.4.5 Determine Required Capabilities Needed to Address the Situation

Once the root causes or contributors to a situation are known, the methods to correct them and/or take advantage of opportunities can be specified. For simple process improvement situations, it may only be necessary to recommend process changes without adding new capabilities or other resources.

For more complicated situations and for opportunities, new capabilities may be needed, such as, software, machinery, skilled staff, or physical plants or properties. The business analyst will recommend suitable capabilities based on discoveries made during the root cause analysis or based on the concepts and contributors to success that were identified when analyzing an opportunity.

Various methods exist for determining new capabilities. When formal root cause analysis is not performed, business people may "jump to solutions" to solve their perceived problems. The result is often adding new capabilities that address only part of a situation and may also be more expensive than needed.

2.4.5.1 Capability Table

By examining each problem and the associated root causes, a needed capability can be discovered. One tool to help facilitate this analysis is a capability table, where the business analyst lists each limiting factor or problem, specifies the associated root causes, and then lists the capability or feature required to address the problem.

Capabilities may also be represented in a visual model like a feature model, see Section 4.10.3 for more information.

Example—Table 2-3 provides a capability table for the insurance company example.

Table 2-3. Capability Table Example

Problem/Current Limitations	Root Cause(s)	New Capability/Feature
Insurance claims too slow	Limited claims agents	• Additional trained agents • Higher pay for claims agents
	Necessity for in-person inspections	• New policy for skipping examinations based on insured's policy length, claim history, and initial intake interview • Use of insured's technology to record damage
	Limited claims adjusters	• Additional trained adjusters • See Section 2.2
Reports delayed	Limited ways to accept/record claims and reports	• New methods of accepting claims • Submit reports remotely
	Paper reports	• Eliminate paper reports

2.4.5.2 Affinity Diagram

Another method for determining capabilities is to use an affinity diagram. Affinity diagrams show categories and subcategories of ideas that cluster or have an affinity to each other. For problem solving, an affinity diagram is used to help organize and structure major cause categories and organize them by the capabilities needed to solve them.

Affinity diagrams are used for multiple purposes. These are handy for drawing out common themes when faced with several disparate and potentially unorganized findings. These diagrams can be used to help connect related issues of a problem or opportunity. They can also provide insights into collections of factors to help understand root causes and possible solutions to problems. The latter point makes them useful for generating the necessary capabilities to address a problem or opportunity.

Example—Figure 2-7 shows how related causes and capabilities can be organized for the insurance company example.

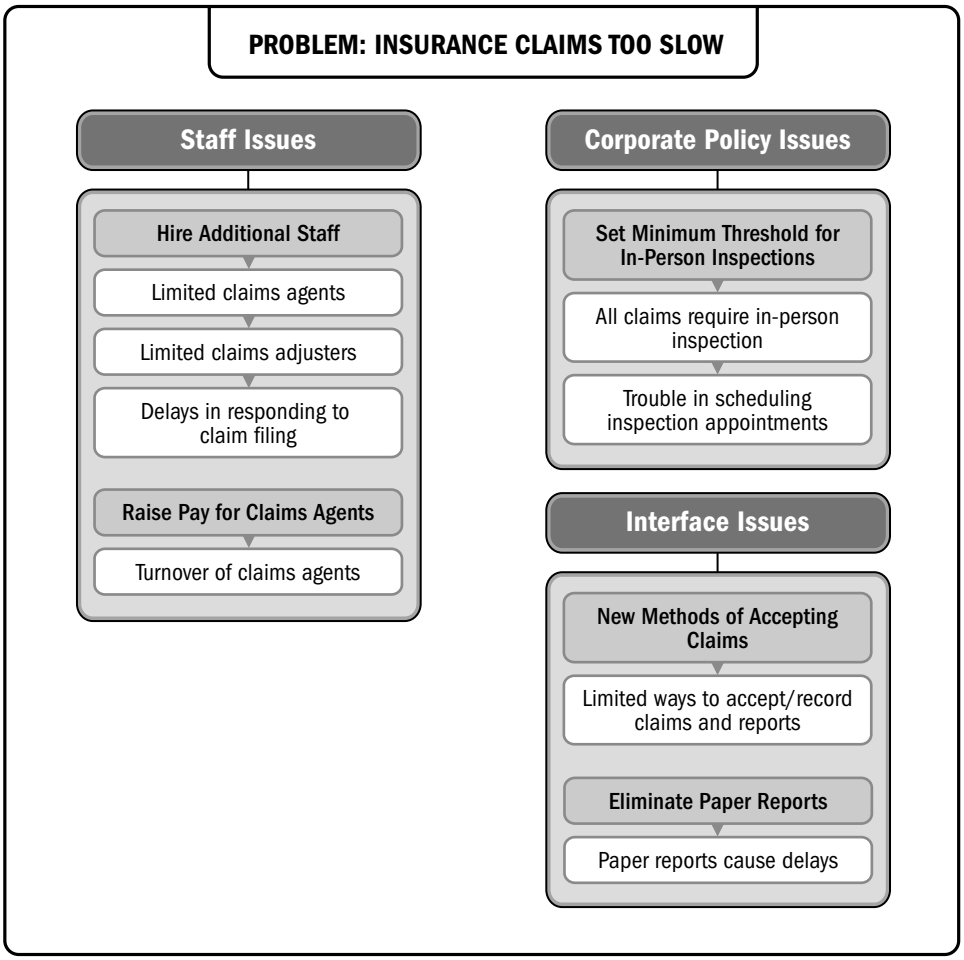


Figure 2-7. Affinity Diagram Example

2.4.5.3 Benchmarking

Another way to determine new capabilities is to conduct benchmarking studies of external organizations that solved similar problems or seized opportunities that the organization is considering. Benchmarking studies often guide final recommendations to address the situation as well as highlight which recommendations not to make.

Stakeholders are almost certain to be interested in what the competition is doing, so it is always helpful to have answers about the competition when those questions arise. Benchmarking is also common in noncompetitive situations; for example, when one governmental agency or jurisdiction shares a solution approach with other groups.

A competitive analysis study typically falls into one of three types:

- Casual review of other organizations' public documents such as corporate reports or web pages,
- Study of the major variables factoring into a solution or "secret shopping" at competitors' stores or websites, or
- Reverse-engineering a specific product to determine its component details and the capabilities needed.

Benchmarking, like competitive analysis, provides insights into how other organizations are responding to the same challenges experienced by the performing organization. Benchmarking of capabilities provides insights that the organization would not have thought of on its own or would have used valuable time to discover them by trial and error.

> *Example*—The insurance company could discover through benchmarking that a competitor offers video chat sessions between customers and their claims adjuster. This discovered capability should be scrutinized to determine the amount of value it would add to the potential solution under consideration.

2.4.6 Assess Current Capabilities of the Organization

Once it is known which capabilities are needed to correct a situation, it is necessary to determine the organization's current capabilities. The organization may have the necessary capabilities to resolve a situation without needing to add new ones. For example, an organization may already have the resources or capabilities needed to solve a problem through process improvement or staff reorganization. In those cases, the solutions tend to be simple, although not always easy to accomplish.

However, when a formal needs assessment is performed as described in this section, it is likely that the needed capabilities do not exist. Before recommending new capabilities, the business analyst should identify and assess current capabilities as they relate to the situation.

Common methods for assessing the current capability state include, but are not limited to the following:

- **Process flows.** Performing "as-is" process analysis or reviewing existing models reveals those current processes in place that could be refined or extended to fulfill the new capabilities. They also uncover resources that could potentially be redeployed to provide new capabilities. The absence of existing processes needed to solve a problem could reveal a larger disparity to fill.
- **Enterprise and business architectures.** Enterprise and business architectures are methods to describe an organization by mapping its essential characteristics such as people, locations, processes, applications, data, and technology. When a formal architectural framework is in place, it can be reviewed to provide a current capability inventory. As with any documentation, the architecture may not reflect reality and

the current capability may well already be in place. Therefore when using existing documentation, effort needs to be made to validate that the models are up to date and still valid.

- **Capability frameworks.** A capability framework is a collection of an organization's capabilities, organized into manageable pieces, similar to business architecture. When this framework exists, it can be reviewed to provide a current capability baseline.

  Example—An insurance company has current-state process flows for the claims processes such as the one shown in Figure 2-6. The company may also have some of the enterprise architecture in place, which may be useful even when models are at a high level. The process flows may need to be updated to reflect current processes, and the architecture components need to be revised to provide greater detail, but both serve as a starting point.

2.4.7 Identify Gaps in Organizational Capabilities

After identifying needed capabilities and assessing current capabilities related to a given situation, any gaps or missing capabilities that exist between the current and needed states are the capabilities that need to be added. These capabilities are commonly referred to as the "to be" features and functions and are easily identified by performing this gap analysis. Gap analysis is the technique of comparing the current state to the future state to identify the differences or gaps.

Example—In the insurance company example, at this stage of the needs assessment, the business analyst is analyzing the current capabilities to the list of needed capabilities in order to identify the capabilities that are missing to achieve the desired future state. The needed capabilities that are required to solve the business problem become the rationale for creating a program or project. Table 2-4 shows an updated capability table with the deliverables added that will fill the gaps.

Table 2-4. Capability Table with Gaps Listed

Problem/Current Limitations	Root Cause(s)	New Capability/Feature	Project Deliverables to Fill Gaps
Insurance claims too slow	Limited claims agents	• Additional trained agents • Higher pay for claims agents	• New training program for agents • Increased hourly rate for claims agents
	Necessity for in-person inspections	• New policy for skipping examinations based on insured's policy length, claim history, and initial intake interview. • Use of insured's technology to record damage	• Create new corporate policy with thresholds for visits • Develop new processes and interface to import insured's pictures of damage
	Limited claims adjusters	• Additional trained adjusters • New processes and interfaces to facilitate claims	• New training program for adjusters • Develop new processes and interface to import insured's pictures of damage
Reports delayed	Limited ways to accept/record claims and reports	• New methods of accepting claims • Submit reports remotely	• Develop new interface for online acceptance of claims • Develop new interface to submit remote claims reports
	Paper reports	• Eliminate paper reports	• Create online and mobile reports

2.5 Recommend Action to Address Business Needs

Once the gaps between the current and needed capabilities are identified, a recommendation can be made to fill the gaps. The needed features and functions are only part of the recommendation, though. The business analyst also provides a high-level approach to add the new capabilities, alternative approaches to consider, the feasibility of each alternative, and a preferred order to the alternatives. This work provides significant value to the organization as it helps to ensure that the organization does not implement the wrong solution.

2.5.1 Include a High-Level Approach for Adding Capabilities

A complete recommendation includes a high-level proposal stating how the needed capabilities will be acquired. This approach is not a detailed project management plan and does not include the level of detail in a project charter. Instead, it is a suggested path for adding the capabilities. The business analyst should solicit preliminary feedback from business and technical architects when the recommendation is going to include new or modified hardware or software. For complex situations, when developing the approach, the business analyst should work with the architects to develop a high-level view of how the new capabilities will be interfaced with existing systems and applications, noting any major dependencies that will exist when the new capabilities are added.

> *Example*—Using the insurance company example, additional security infrastructure may be needed to allow claims to be submitted from mobile devices or to allow video chats with claims adjusters. Dependencies require time and money to implement before the desired capabilities can be added and will affect the net benefits of the chosen solution.

2.5.2 Provide Alternative Options for Satisfying the Business Need

There is rarely only one potential solution to a business problem. While there are often multiple approaches for adding new capabilities, a recommendation should include all of the most viable options. The standard build vs. buy alternatives are common options, and so are combinations of the two, in addition to alternative vendors to choose from. The business analyst evaluates and analyzes all the information gathered during the needs assessment to produce multiple possible solutions.

The primary reason for providing alternatives is to show that the alternatives were considered and to forestall objections from those who favor them. Another reason for including alternatives is because the preferred approach may not be acceptable. Constraints such as cost, schedule, staffing, or vendor bias may preclude an option from being chosen, therefore, potentially, one alternative will be an acceptable choice.

While the business analyst does not decide which solution may be the best one, the business analyst is usually expected to make a recommendation and to support the recommendation with facts and evidence. The solution decision is made primarily by the business sponsor or problem owner with additional input about solution feasibility from the solution team or those responsible for developing the solution.

2.5.3 Identify Constraints, Assumptions, and Risks for Each Option

Understanding the list of constraints, risks, and assumptions is useful when analyzing project proposals for addressing the business need and when conducting the project planning should the proposal be accepted.

> ***Collaboration Point***—Business analysts may draw on a project manager's expertise in risk planning and mitigation. Both roles are concerned with risk management. Project managers are focused on assessing the risks of the project as a whole, while the business analyst is focused on assessing the product risks. Business and technical architects can support this work too, as they may help uncover additional constraints or opportunities. Examples of this are the risks of a solution option or the risks associated with a particular portion of the business analysis process.

2.5.3.1 Constraints

Constraints are any limitations on a team's options to execute a program or project and may be business- or technical-related. It is important to assess any known constraints regarding a potential solution when generating alternatives. For example, if a time constraint to deliver a solution is imposed on a potential project, a custom-built solution may not be a viable option and would need to be removed or, at best, not be listed as a preferred option. Design and implementation constraints are important to investigate early in the process as well.

2.5.3.2 Assumptions

Assumptions are factors that are considered to be true, real, or certain, without actual proof or demonstration. Recommendations, proposals, and business cases are projections of the future, based on information limited to the person or team compiling them. As such, a number of assumptions are typically made, subject to the information available to the team. The major assumptions should be documented so that decision makers are aware of them when they evaluate the recommendations. If any assumptions are proven to be incorrect, it is important to reassess the identified needs or perhaps the entire business case.

2.5.3.3 Risks

Risks are uncertain events or conditions that may have a positive or negative effect on one or more project objectives if they occur. Potential risks for various alternatives can have a significant effect on the perception of business leaders when selecting potential projects. Major risks for each alternative should be noted along with potential mitigation strategies and their costs. Any risks that need a response may add cost to a potential solution.

2.5.4 Assess Feasibility and Organizational Impacts of Each Option

Before deciding which option is preferred, the business analyst assesses the feasibility of each potential option. In doing so, one or more options may be discarded due to their lack of feasibility. The assessment involves comparing potential solution options for how viable each appears to be on key variables or "feasibility factors"

(explained below). This comparison along with the elimination of any options that are not deemed sufficiently feasible is called feasibility analysis.

Another reason for analyzing the feasibility of alternatives is to reserve the often painstaking work of cost-benefit analysis for only the most feasible option. The type of feasibility analysis described here is not a full business case in that the latter includes a complete economic analysis. Performing a complete business case only for the most viable options will save valuable time.

Typically, there is not just one factor, such as cost or time, that will determine whether an option is feasible or not. Feasibility is best analyzed according to a variety of important factors. Those factors and the questions to address are as follows:

2.5.4.1 Operational Feasibility

Operational feasibility represents how well the proposed solution fits the business need. It also encompasses the receptivity of the organization to the change and whether the change can be sustained after it is implemented. Meeting the business needs is the most important factor of all, because if an option does not solve the problem in question, the other factors are irrelevant. Sample questions to ask include:

- How well does the option in question meet the business needs?
- How does the solution fit into the organization, including its impact on the organization?
- How well do potential options meet the nonfunctional requirements, for example, sustainability, maintainability, supportability, and reliability?

2.5.4.2 Technology/System Feasibility

Technology feasibility pertains to whether or not the technology and technical skills exist or can be affordably obtained to adopt and support the proposed change. In addition, it includes whether the proposed change is compatible with other parts of the technical infrastructure. Questions to ask include:

- Does the technology for a potential solution exist in the organization?
- If not, is it feasible to acquire it?
- Does the organization have the technical expertise to install or operate the potential solution?

2.5.4.3 Cost-Effectiveness Feasibility

Before conducting a full-fledged cost-benefit analysis of a potential solution, a high-level assessment of the financial feasibility of potential solutions is essential. Cost-effectiveness during feasibility analysis is not intended to be a complete cost-benefit analysis, but an initial and high-level feasibility estimate of costs and value of the benefits. The full cost-benefit analysis is later performed on the most viable option(s) during business case preparation. Potential questions to ask include:

- What are the up-front and ongoing costs for the proposed solutions?

- Are the potential solutions affordable in view of the expected benefits?
- Can the funding be obtained for the investment needed?

2.5.4.4 Time Feasibility

A potential solution will be feasible if it can be delivered within time constraints. Potential questions to ask include:

- Can a given solution option be delivered to meet the organization's time frame?
- How reasonable is a proposed option's timetable?
- When a potential solution cannot be completely delivered by a certain deadline, is it acceptable to deliver it in stages and still meet business needs?

2.5.4.5 Assess Factors

The business analyst assesses the feasibility factors of each prospective solution option to determine how well these options contribute to the goals and objectives assessed earlier. In other words, the feasibility analysis uses the factors as criteria for judging how well each potential option solves the problem at hand and contributes to achieving the goals and objectives. Instead of working alone, it is important for the business analyst to also elicit, document, and leverage the judgments of affected stakeholders whenever possible.

> *Example*—In the insurance company example, one of the needed capabilities is to use their policyholders' technology to record damage. To add the capability, the company could install a commercial off-the-shelf (COTS) package, develop the software in-house, or outsource the development. All three options may be viable and feasible choices.

2.5.5 Recommend the Most Viable Option

After examining potential options for addressing a business need, the business analyst needs to recommend the most viable option. Assuming that more than one option remains viable after feasibility analysis, the business analyst should recommend the most feasible option. If only one option is judged to be feasible, that option, in most cases, would be recommended. When there are no viable options to address the need, one option is to recommend that nothing be done.

When faced with two or more feasible options for solutions, the remaining choices can be rank-ordered based on how well each one meets the business need. For example, when an organization is reengineering its operations as opposed to outsourcing it, these two options can be ranked according to how well each one solves the business problems and contributes to business objectives.

2.5.5.1 Weighted Ranking

Ranking two or more options can be done using various techniques. A practical and effective method is to use a weighted ranking matrix. A weighted ranking matrix or table combines pair-matching with weighted criteria to add

objectivity to a recommendation. Pair-matching is performed by taking each option and comparing it one by one to all other options, and then voting or ranking which option is the most preferred. Weighted ranking is also useful to test an initial or intuitive choice against other options. The criteria used for ranking should align with the goals and objectives identified earlier in the needs assessment.

The basic approach is to select weighted criteria for each item to be ranked. Each option is ranked by voting on it against every other option, one at a time. Scores for each alternative are multiplied by the weights and added to arrive at the score for each option and the overall rankings. Guidelines for constructing and using a weighted ranking table are:

- Include between three and nine criteria as a practical range for the number of criteria. If a problem has fewer than three criteria, then a weighted ranking matrix is rarely needed to analyze each option. If there are more than nine criteria, it will be difficult for stakeholders to judge and compare the alternatives. Additionally, the problem may be overly complex and may need to be broken into subsets to properly analyze it.
- Assign weights by either percentages or decimals. It is usually preferable if the weights add up to 100% or 1.0. Stakeholders performing the ranking may prefer that each criterion have a weight between 1 and 100%.
- Determine how voting will be conducted. Voting for each compared pair of alternatives can be done with a simple "majority rules." Another alternative is to record the number of stakeholders who vote for each pairing. This method provides one vote per person instead of one vote for the winner of the pairing. Teams may prefer to provide the stakeholders who have more authority in the organization with more votes. For example, a sponsor might receive two votes, and everyone else receives one vote.
- Check to ensure that the results make sense to the voters, since unrealistic results will not be helpful. Options that score excessively high or low could indicate skewed results or could indicate that one option is the clear favorite. Ask stakeholders for their subjective responses to the voting outcome. If the results are not trusted by the team, it usually means that the criteria, weights, or the voting method needs to be refined.

  Example—One of the goals of the insurance company is to raise revenues. One criterion would be to compare alternatives against how well each option helps to increase revenue.

See Table 2-5 for an example of a weighted ranking matrix.

Table 2-5. Weighted-Ranking Matrix Example

Items to be Ranked	Criteria (Weight)			Cost (Weight 0.1)	Total Votes	Final Rank
	Increase Revenues (Weight 0.3)	Decrease Claims Costs (Weight 0.4)	Ease of Implementation (Weight 0.2)			
XYZ software package	1 × 0.3 = 0.3	1 × 0.4 = 0.4	1 × 0.1 = 0.2	2 × 0.1 = 0.2	1.1	2
Develop smartphone interfaces in-house	2 × 0.3 = 0.6	2 × 0.4 = 0.8	. . .	. . .	1.4	1
Outsource interface development	. . .	. . .	2 × 0.2 = 0.4	1 × 0.1 = 0.1	0.5	3

Collaboration Point—When constructing a weighted-ranking matrix, the business analyst should consult with the sponsor of the needs assessment to determine which stakeholders to include in the voting process. The stakeholders who are voting should be consulted about which criteria and weights to use. Likewise, decisions regarding the weighting system and the weights require collaboration with stakeholders.

2.5.6 Conduct Cost-Benefit Analysis for Recommended Option

Before recommending a preferred option, a cost-benefit analysis should be performed. The expected project benefits and costs need to be articulated in greater detail during a cost-benefit analysis than during a feasibility analysis. The earlier estimates performed during the feasibility analysis are now replaced by rough order-of-magnitude estimates of benefits and costs. If the business case is accepted and a project is initiated, these estimates will be used during project initiation.

Organizations often have standards that dictate when and how to perform a cost-benefit analysis, including which financial valuation methods to employ. Depending on the organization, consult with a financial analyst or a representative from finance to prepare the cost-benefit analysis to support the business case work. The most common valuation techniques are briefly described here and the recommendation should contain at least one of them.

2.5.6.1 Payback Period (PBP)

The payback period (PBP) is the time needed to recover a project investment, usually in months or years. The longer the PBP, the greater the risk. Some organizations set a threshold level that requires payback periods to be less than or equal to a certain period of time.

2.5.6.2 Return on Investment (ROI)

The return on investment (ROI) is the percentage return on an initial project investment. ROI is calculated by taking the projected average of all net benefits and dividing them by the initial project cost. This valuation method does not take into account ongoing costs of new products or services, but is still a widely used metric. Organizations often have "hurdle rates" that an expected ROI needs to exceed before a project is considered for selection.

2.5.6.3 Internal Rate of Return (IRR)

The internal rate of return (IRR) is the projected annual yield of a project investment, incorporating both initial and ongoing costs. IRR is the estimated growth rate percentage that a given project is expected to attain. Hurdle rates are often established, such as a projected IRR, and need to exceed a particular level in order for a project investment to be considered.

2.5.6.4 Net Present Value (NPV)

The net present value (NPV) is the future value of expected project benefits expressed in the value those benefits have at the time of investment. NPV takes into account current and future benefits, inflation, and it factors in the

yield that could be obtained through investing in financial instruments as opposed to a project. Any NPV greater than zero is considered to be a worthwhile investment, although a project with an NPV less than zero may be approved if the initiative is a government mandate, for example.

The expected benefits listed in the cost-benefit analysis will be valuable measures to use when evaluating the outcomes of the program or project. During needs assessment, the business analyst may highlight key evaluation measures as part of the recommendation. If these measures are not captured during needs assessment, these are documented during business analysis planning, which is discussed in Section 3.

> ***Collaboration Point***—When estimating benefits and costs, a business analyst may work with a project manager with expertise in preparing estimates. Business analysts should also seek the support of financial analysts within the organization who can assist with the application of the valuation methods against the alternatives.

2.6 Assemble the Business Case

Not all business problems or opportunities require a formal business case. Executives in an organization may approve programs and projects based on competitive pressure, government mandate, or executive inclination. In those cases, a project charter to initiate a program or project is sufficient.

In most instances, the analysis performed in the business case helps organizations select the best programs and projects to meet the needs of the business. Business cases help organizations scrutinize programs and projects in a consistent manner. When this process is embraced, organizations will consistently make better decisions.

As described previously, a business case explores the nature of the problem or opportunity, determines its root causes or contributors to success, and presents many facets that contribute to a complete recommendation.

Organizations often have their own standards for what to include in a business case and will employ a set of templates or business case software to simplify and standardize the process. A common set of components in any business case should minimally include the following:

- **Problem/Opportunity.** Specify what is prompting the need for action. Use a situation statement or similar way to document the business problem or opportunity to be accrued through a program or project. Include relevant data to assess the situation and identify which stakeholders or stakeholder groups are affected.
- **Analysis of the Situation.** Organizational goals and objectives are listed to assess how a potential solution supports and contributes to them. Include root cause(s) of the problem or main contributors of an opportunity. Support the analysis through relevant data to confirm the rationale. Include needed capabilities versus existing capabilities. The gaps between them will form the program or project objectives.
- **Recommendation.** Present results of the feasibility analysis for each potential option. Specify any constraints, assumptions, risks, and dependencies for each option. Rank-order the alternatives and list the recommended one; include why it is recommended and why the others are not. Summarize the cost-benefit analysis for the recommended option. Include the implementation approach, including milestones, dependencies, roles, and responsibilities.

- **Evaluation.** Include a plan for measuring benefits realization. This plan typically includes the metrics to evaluate how the solution contributes to goals and objectives. It may necessitate additional work to capture and report those metrics.

  Note: The formality of a documented business case and project charter is commonly required in large or highly regulated companies. While this formality may not always apply in smaller organizations or in some organizations which have adopted an agile approach, the thought process of defining the problem/opportunity, analyzing the situation, making recommendations, and defining evaluation criteria is applicable to all organizations.

2.6.1 Value of the Business Case

When a business case is created, it becomes a valued input to project initiation, providing the project team with a concise and comprehensive view of the business need and the approved solution to that need. More than a simple input, a business case is a living document that is constantly referenced throughout a program or project of work. It may be necessary to review and update a business case based on what is discovered as a program or project progresses over time.

When a business case is inadequate or nonexistent, the product scope may be unclear or poorly defined. This in turn often leads to scope creep, which results in rework, cost overruns, and project delays. A business case can help to address the possible risks of having to cancel a project due to loss of sponsor or stakeholder support, costs exceeding the perceived benefits, and changes to the business. Possibly worse than terminating a project is finishing a project only to have the end-product not be used because the solution did not match the business needs. The net effects of canceled projects and unused products are wasted investments, lost opportunity costs, and frustration.

> ***Collaboration Point***—Business analysts work closely with the sponsor to create a business case. When the project manager is known, the business analyst consults with the project manager to achieve a stronger business case through close collaboration. The project manager may better understand the business need, feasibility, risks, and other major facets of the business case. When a project or program is approved, the business analyst and project manager may work together during initiation to ensure the business case is properly translated into a project charter and/or similar document.

3

BUSINESS ANALYSIS PLANNING

3.1 Overview of this Section

Within business analysis, planning consists of the activities that are performed in order to ensure that the optimal business analysis approach is selected for the project and that:

- Stakeholders are thoroughly identified and analyzed;
- Business analysis activities and deliverables are defined and agreed to;
- Processes that will be used for validating, verifying, and approving requirements and solutions are acceptable to key stakeholders;
- The process for proposing changes to requirements is defined and understood; and
- Key stakeholders are aware of and support the activities and time commitments required to complete the requirements effort.

The business analysis approach is simply the method the business analyst uses when managing and performing the business analysis activities on the project. As described later in this section, the approach is described within the business analysis plan.

How business analysis planning is conducted is heavily dependent on the selected project life cycle; therefore when a planning activity is performed differently across life cycles or not performed at all, the differences are noted in this section. There is no one approach to business analysis planning that will work for every project, so ultimately the business analyst should understand the context and project characteristics enough to ensure the planning activities are sufficiently sized for the situation.

This section describes the important things one needs to think about when defining the business analysis approach. The level of thought described in this section applies to all programs and projects whether it is performed all at once at the forefront of the project or throughout the project, regardless of the degree of formality used to document the business analysis plan.

3.2 The Importance of Business Analysis Planning

Planning the business analysis work is critical for project success. When business analysis planning is bypassed, it is difficult to understand the scope of work, stakeholder's expectations, and the appropriate amount and level of business analysis required for the project. This in turn makes the estimation process difficult and can result in unrealistic expectations by those involved in the requirements-related activities. Business analysts who begin

elicitation sessions without a well thought out roadmap of how they will address the work will often find themselves pressed for time and rushing through activities to meet a schedule to the detriment of the project.

3.2.1 Rationale

Because requirements are the foundation from which the project is based and a key contributor to project success, the sponsor and project manager should ensure that a sufficient level of business analysis planning is conducted. Many projects are initiated with tight timelines that place pressure to address the tactical activities before the plan. The project team should avoid the urge to rush into requirements elicitation without first understanding the expectations for the business analysis process and the roadmap for pursuing the work.

Business analysis planning achieves the following:

- Sets expectations with the sponsor, project team, and key stakeholders as to the business analysis activities that will be performed;
- Ensures that roles are identified, understood, and communicated to everyone participating in the business analysis process;
- Achieves buy-in and support for the business analysis process before work begins;
- Provides context to support estimation of the business analysis activities; and
- Produces a more efficiently run business analysis process, because activities are not missed or excessively performed.

While planning provides many benefits and reduces a number of requirements-related risks, planning work should be judiciously performed to ensure the process is not too heavy or formal for the needs of the program or project. It may not be necessary or advisable to plan entirely up-front and in great detail on every type of program or project. When using a predictive life cycle, planning is performed up-front prior to elicitation. In adaptive life cycles, some planning is performed up-front and plans are adapted or evolve over the course of the program or project. Too much planning can be counterproductive, therefore the business analyst needs to plan a sufficient level of detail to address the specific needs of the project such as the size, complexity, and risk level.

3.2.2 Business Analysis Planning and Project Management Planning

Business analysis is a critical portion of the overall project activities. The work involved to perform a successful business analysis process is detailed and the number of activities conducted can be quite extensive. Program and project success is dependent on adequate business analysis; therefore, attention should be given to ensure that activities are well thought out and meet the needs of the program or project and the stakeholders.

Business analysis planning and scheduling is not performed independent of project management scheduling activities. It is a best practice to have the project manager and business analyst working closely together while the business analysis approach and plan is formulated. Business analysts will develop a work plan to cover the activities they are responsible for performing; however, the work plan should be integrated into the overall project management plan managed by the project manager.

3.3 Conduct or Refine the Stakeholder Analysis

A stakeholder is an individual, group, or organization who may affect, be affected by, or perceive itself to be affected by a decision, activity, or outcome of a program or project. Stakeholder analysis is a technique used to systematically gather and analyze quantitative and qualitative information to determine whose interests should be taken into account throughout the project.

Stakeholder analysis is often conducted during the planning phase so that the project team can understand the stakeholder impacts and influences on the business analysis process as early as possible. Stakeholder analysis is performed iteratively and is revisited throughout the project as new stakeholders are discovered or existing stakeholders are determined to no longer be impacted by the proposed solution. Early planning will produce the initial stakeholder list, but further stakeholder identification will maintain the list. When there are a large number of stakeholders identified, stakeholder analysis may involve grouping the list by common characteristics which helps to streamline analysis.

Project managers use stakeholder analysis to assess how the stakeholder groups will influence and impact the project work. Business analysts use the results to understand how the stakeholders will impact the business analysis process. The business analyst considers a number of stakeholder characteristics before determining how to best conduct the business analysis activities. Both roles assess stakeholders to understand how to communicate, collaborate, manage, and set expectations.

> ***Collaboration Point***—How project managers and business analysts collaborate and work together on stakeholder analysis varies by organization, but it is more efficient if they work collaboratively with their project and product teams to avoid duplication of effort. Collaboration will also provide a more insightful examination of stakeholders as the experiences and expertise of all participants are leveraged to identify project and product impacts.

3.3.1 Techniques for Identifying Stakeholders

There are various techniques that can be used to uncover and analyze stakeholders. Using a variety of techniques helps to draw out the information from different perspectives and angles. The discovery approach may be as simple as asking other stakeholders for input. Existing documentation, such as organizational charts or process flows, can help to identify user groups. Common techniques that can be used in the discovery of stakeholders are brainstorming, decomposition modeling, interviews, surveys, or organizational modeling, to name a few.

3.3.1.1 Brainstorming

Brainstorming is a data gathering technique that can be used to identify a list of ideas in a short period of time (e.g., list of risks, stakeholders, or solutions to issues). Brainstorming is conducted in a group environment and is led by a facilitator. A topic or issue is presented and the group is asked to generate as many ideas or solutions as possible about the topic. Ideas are provided freely and rapidly and all ideas are accepted. Because the discussion occurs in a group setting, participants feed off of each other's inputs to generate additional ideas. The responses are documented in front of the group so progress is continually fed back to the participants. The facilitator takes

on an important role to ensure all participants are involved in the discussion and to ensure no one individual monopolizes the session or critiques or criticizes the ideas that are offered by others.

Brainstorming is comprised of two parts: idea generation and analysis. The analysis is conducted to turn the initial list of ideas into a usable form of information. In business analysis planning, brainstorming can be leveraged to build the initial list of stakeholders, to discover new stakeholders, or to identify a list of tasks to be included in the business analysis work plan.

3.3.1.2 Organizational Charts

An organizational chart helps with stakeholder discovery. The business analyst reviews the chart to locate stakeholder groups who may be impacted by the product or service. This may include departments who operate or maintain a system, produce a product or service, support customers, or influence product or service decisions within the area under analysis.

Whether existing organizational charts are used as the starting point or the charts are built from scratch, the work to finalize the chart is completed through a series of discussions with the managers of the departments being modeled. Based on the size of the organization and how the organizational charts are being leveraged during the analysis, the business analyst determines whether it makes sense to take a role organizational chart down to the individual stakeholder level. If the goal is to only identify the number of groups impacted by the project, the role organizational chart may be the sufficient level of detail required.

The business analyst should keep in mind that roles may be conducted differently across the organization and may vary regionally or by type of customer supported. Additionally, stakeholders from the same group may use a product or service differently. When differences are identified during analysis, the model should be updated to reflect a role for each of the variations discovered. The ultimate goal is to uncover all of the stakeholders who have needs to be met by the product or service and may have requirements to provide for the product or service. An oversight of just one role type can result in implementing a solution that fails to meet the needs of hundreds or even thousands of customers.

3.3.2 Determine Stakeholder Characteristics

After developing or refining the stakeholder list, the business analyst analyzes the characteristics for the stakeholders identified. The list of characteristics is almost endless, so the analyst will choose those that have the most significance or most relevance to the project. Some commonly applied characteristics along with a brief explanation of their importance and significance for consideration are as follows:

3.3.2.1 Attitude

Consider analyzing stakeholder attitudes to identify which stakeholders support the project and proposed solution and which stakeholders do not. Stakeholders who are positive about the project and solution may serve as project champions; these stakeholders will assist the project team in building excitement and support for the solution. Identifying nonsupportive stakeholders can help identify areas where additional collaboration is needed. Spending

more time with these stakeholders may uncover unspoken business needs, requirements, training issues, resource constraints, or past and current experiences important for the project team to understand. Uncovering stakeholder likes and dislikes may even bring to the forefront unspoken concerns about the proposed business analysis process.

Attitudes are not solely based upon likes and dislikes for the project. A stakeholder who exerts dislike or disinterest in a project may simply see no value in the initiative or final solution. The stakeholder group may be included in the project because its business processes are impacted, but these stakeholders may not be recipients of direct value from the work. Stakeholder groups may be recipients of additional work after a project is implemented and therefore never display a high level of interest. Those stakeholders whose workload is likely to increase may never become supportive of the project. Understand the concerns of the disinterested stakeholders and look for ways to obtain their engagement despite the lack of support the stakeholder may have toward the project.

3.3.2.2 Complexity

A stakeholder group can be considered complex for a number of reasons, including but not limited to whether the group

- Is comprised of a large number of stakeholders,
- Is made up of stakeholders with vastly different needs,
- Has a number of business processes impacted by the project,
- Exhibits a lack of uniformity across business processes,
- Interacts with a number of business units to complete their work, or
- Performs work across a number of IT systems, and/or systems external to the organization.

Understanding complexity levels will help when quantifying and planning the number of requirement sessions to conduct, when determining the right amount of requirements-related documentation to produce, and when determining the level of formality to apply in those deliverables. Complexity levels are also helpful to understand when assessing solution options and the change impacts that a project will have on stakeholder groups.

PMI's *Pulse of the Profession® In-Depth Report: Navigating Complexity*[5] identified multiple stakeholders, followed by ambiguity in project features as the top reasons for project complexity (see Figure 3-1). By understanding the factors that drive complexity on projects and among various stakeholder groups, the business analyst is able to more effectively plan the best approach for conducting the business analysis work, including selecting the best techniques for eliciting and analyzing the product requirements.

> ***Collaboration Point***—Project managers are interested in understanding the complexity level of the project in order to allocate sufficient time for project activities, including collaboration and communication with stakeholders. The business analyst assesses complexity to understand how best to approach business analysis activities and to understand the impact that the change will have on stakeholders.

[5] Project Management Institute. (2013). *PMI's Pulse of the Profession® In-Depth Report: Navigating Complexity.* Available from http://www.pmi.org/~/media/PDF/Business-Solutions/Navigating_Complexity.ashx

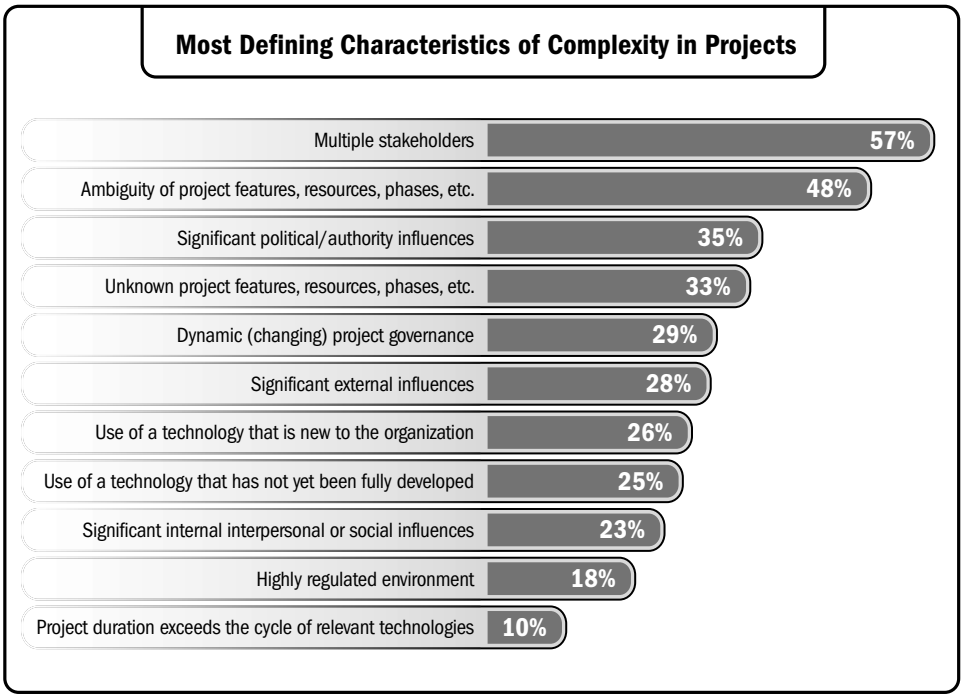


Figure 3-1. Understanding Complexity (Source: *PMI's Pulse of the Profession® In-Depth Report: Navigating Complexity*)

3.3.2.3 Culture

Consider the cultural diversity that exists among stakeholders and make necessary adjustments to the business analysis process to ensure these differences are considered. Many aspects of culture may be worthy of analysis, including age, nationality, or group, departmental, or organizational culture. Culture can impact how the business analyst proposes communicating with stakeholders, eliciting requirements, conducting requirement walkthroughs, and running the prioritization, approval, and change processes. Culture and location are not the same thing. Even when all the stakeholder groups reside in one location, there are cultural differences among the groups. Cultural differences impact how stakeholders:

- Perform their work,
- Interact with other team members,
- Contribute to the decision-making process,
- Interpret nonverbal communication,
- Understand the primary written and spoken language of the team,
- Question or interact with authority,
- View their role on the team,

- Raise questions or issues,
- Negotiate, and
- Deal with conflicts.

A business analyst who takes the time to recognize and understand the cultural differences will gain an awareness and appreciation for the diversity and can use their understandings to run a more effective business analysis process.

3.3.2.4 Experience

Understanding the experience level of the project stakeholders provides helpful information for planning activities. Elicitation may be completed more efficiently when stakeholders have prior experience participating in requirement workshops or validating requirements than with a team of stakeholders who are unfamiliar with business analysis or do not understand the value of the business analysis activities. A business analyst should consider the number of years of industry experience stakeholders have, whether they have worked with the organization for an extended length of time or whether an individual stakeholder is new to the organization. These factors can influence how requirements are validated and approved as well as the number of stakeholders to engage from a stakeholder group. The business analyst will also find it important to analyze experience across stakeholder groups to ensure that a sufficient breadth of business knowledge is represented on the requirements team. Where gaps are identified, the business analyst, working with the project manager, should seek to acquire additional resources to provide the needed or missing experience for the requirements-related activities.

3.3.2.5 Level of Influence

Understanding the amount of influence a stakeholder has within an organization helps to identify where influence can serve as a motivator as much as it can serve to distract or deter others from embracing the work of the project team. A person's level of influence is often tied to his or her position within the organization; however, influence is not solely associated to a person's rank, reporting structure, or job title. A person's level of influence is also affected by business relationships, reputation, knowledge or level of experience, or successes within the organization. Analyze the level of influence to understand the power an individual or stakeholder possesses. This analysis will highlight those who can help rally for the solution as well as identify those that can hinder project success. Through this analysis, the business analyst will be able to identify influential stakeholders and areas where more time needs to be spent on building relationships and collaborating with key individuals.

3.3.2.6 Location and Availability

A global workforce and an increase in the number of organizations supporting a virtual work environment are two trends that make analyzing location a worthwhile step in stakeholder analysis. Even when the project team is centralized, the business analyst should understand whether remote work is supported. The business analyst should review the stakeholder list and identify where each stakeholder is based and the locations from which each conducts their work. If stakeholders work from multiple locations, then the frequency that stakeholders work from each location should be noted.

When stakeholders work remotely, it is also helpful to identify the methods used for connectivity. This information can be used when determining the best approach for collaboration, for example, conference calls, desktop sharing, web conferencing, etc. Understanding availability, including work hours, and whether a flexible work schedule is used or a four-day work week is in place is beneficial as availability impacts the frequency by which the business analyst can conduct elicitation sessions. The business analyst may choose to elicit requirements through facilitated workshops when key stakeholders are local but use interviews or surveys when key stakeholders are dispersed or when time zone differences make scheduling conference calls difficult. Documentation analysis may be used when stakeholders are not available at all.

Location affects a number of business analysis planning decisions such as:

- Determining the techniques that are most effective to elicit requirements,
- Estimating how much time is required to complete business analysis activities,
- Deciding on the formality and level of detail required for the business analysis deliverables,
- Deciding on the types of models to use,
- Determining the approach and frequency of collaboration, and
- Determining how requirements will be managed, maintained, and shared with project stakeholders.

3.3.3 Techniques for Grouping or Analyzing Stakeholders

Stakeholders can be grouped once their characteristics are understood. Stakeholder lists can quickly become long and difficult to manage; therefore, placing stakeholders into groups will allow for easier management of the information. Groupings can be structured by similar interests, common needs, level of importance, by roles, motivations, complexity level, location or many other qualities. Primary and secondary designations can also be used to separate stakeholders by those who are primarily or directly impacted by the project or solution compared to secondary stakeholders who are indirectly impacted. The business analyst should choose the type of stakeholder grouping that best meets the objectives of the analysis.

There are several techniques that can be used to group or analyze stakeholders such as job analysis, organizational modeling, personas, process modeling, risk analysis, and stakeholder maps (see 3.3.3.1 and 3.3.3.2).

3.3.3.1 Job Analysis

Job analysis is a technique used to identify the job requirements and competencies needed to perform effectively in a specific job or role. The technique is often used when a new job is created or when an existing job is modified.

Organizations use job analysis when drafting job descriptions and when determining the best person to fill a position. The output of job analysis may include a number of details such as the description of the work, a depiction of the work environment, a detailed list of the activities a person will perform, a listing of the interpersonal skills required to perform well in the job, or a list of required training, degrees, and certifications. Job analysis may also be used to determine training needs, formulate information needed to write a job posting, and to support the performance appraisal process.

Business analysts can review the job analysis to understand how particular roles are performed by stakeholders. When the project entails replacing or revising workflow and business processes, the business analyst uses the job analysis to understand how the existing job is currently performed. When the project entails the creation of a new job, job analysis may be used to specify the tasks required for the new role.

3.3.3.2 Persona Analysis

Persona analysis is a technique that is conducted to analyze a class of users or process workers. It is a powerful tool for understanding stakeholder needs and for targeting product design and behavior for each class of user. A persona is a fictional character created to represent a user group or group of stakeholders who have similar needs. During planning activities, the business analyst uses personas to understand the characteristics of various stakeholder groups and can apply that information to select the best business analysis approach to meet the needs of the project.

Personas are often used in IT systems development and product development to help design or map out a user's experience. While it is often not possible to obtain requirements for every stakeholder on the stakeholder list, it becomes necessary to group the list of stakeholders into various user classes and then build a persona for each class. The objective is to analyze usage information or draw out user requirements to determine how a user class interacts with a system or how a user class would use a product.

The difference between a persona and a stakeholder is that the persona includes much more detail about how it interacts with the problem or solution space, such as, device literacy and preferences, preferred method for searching, and frequency of performing specific actions, etc. Generally, a business analyst does not need this kind of in-depth information for all of the identified stakeholders. In stakeholder analysis, types are identified; however, the business analyst also identifies specific individual stakeholders. In persona analysis, the business analyst focuses on types, often giving them a name and picture for fun (e.g., "Paul the Purchaser"). Unlike stakeholders, a persona may or may not have a material interest in the outcome of the project; a persona simply may be a user who has no influence.

A persona is written in the form of a narrative and tells a story about the user class. Personas describe the goals, behaviors, motivations, environment, demographics, and skills of the user class. The information provided in the persona is behavioral in nature versus the information that is obtained from a job analysis that would be descriptive in nature. A persona is usually one to two pages in length.

3.3.4 Assemble the Stakeholder Analysis Results

As stakeholder characteristics are analyzed, the business analyst may choose to document the results of the analysis in a way that can be shared with the project manager, project team, and sponsor. Much of the information being gathered and analyzed could be considered sensitive in nature; therefore, the business analyst should exercise care when distributing the analysis to a broad distribution group.

The intent of analyzing these characteristics is not to stereotype the stakeholders but to obtain a better understanding of them so that the decisions made when formulating an approach to the business analysis work

is optimal and conducive to running a high-performing business analysis process. The additional stakeholder characteristics shared after the stakeholder analysis provides the project team with insights for determining how best to collaborate and work with the stakeholders throughout the entire project life cycle, especially during the business analysis activities.

3.4 Create the Business Analysis Plan

A business analysis plan is created to document how the business analysis process will be performed, including the plan decisions agreed to by the project team. The plan can be constructed formally and be reviewed and approved by the project team or the document can be more informal, simply serving as a record of team decisions. Much value is obtained when building the plan as a team and in collaboration with the stakeholders involved in the business analysis activities.

Plan templates may exist, and when available, business analysts can leverage them as a starting point to avoid building the plan from scratch. Using an existing plan as a starting point also provides the benefit of leveraging existing corporate standards, culturally accepted practices, and any expected governance or management steps expected for the business analysis work.

Not all information may be known at the time the plan is developed; therefore, the business analyst may be required to make some assumptions. The business analyst includes any assumptions made in the business analysis plan document. Making process decisions up-front in the planning phase reduces the risk of being held up debating process during the execution phase.

In some situations, planning in the execution phase may be preferable to planning everything up-front to avoid reaching premature decisions based on incomplete information. In some cases, it may not be possible or desired to plan the entire approach up-front. Rolling wave planning may be used, which involves planning from a high-level first and a more detailed level later on in the project when the activities are ready to be performed. In these cases, the business analysis plan evolves over time and the business analyst reviews planning and approach decisions with the sponsor, project manager, and key stakeholders as more details are known. The business analyst needs to balance the advantages provided by up-front planning against any disadvantages when considering how much up-front planning to perform.

> ***Collaboration Point***—The business analyst should build the business analysis plan collaboratively with key stakeholders to attain buy-in. A plan that is constructed by the team provides a sense of ownership to those involved and sets expectations by bringing awareness to how the work will be performed.

3.4.1 Business Analysis Plan vs. Requirements Management Plan

In the project management discipline, a requirements management plan is a component of the project management plan and describes how the overall requirements of the project will be elicited, analyzed, documented, and managed across the project. The requirements management plan covers planning decisions for both the product and project requirements. Previously in the business analysis discipline, the term requirements management

plan referred to a document that contained planning decisions about how requirements were to be stored and maintained, as well as decisions about baselining, updating, and change impact analysis. The term has evolved in some organizations to also encompass planning decisions for the business analysis process: how requirements will be elicited, analyzed, documented, and managed across the project. Because business analysis is focused on product requirements and the primary focus of this practice guide is the business analysis process, this practice guide will use the term "business analysis plan" to refer to all information that is documented regarding business analysis planning decisions.

> ***Collaboration Point***—The business analysis plan works in conjunction with the requirements management plan. The business analyst should work closely with the project manager to ensure content is not duplicated between the documents and that the planning decisions for both the project requirements and product requirements are covered.

3.4.2 What to Include in the Business Analysis Plan

The business analysis plan is focused on the scope of the business analysis effort. This includes a list of the activities to be conducted and the business analysis deliverables to be produced. A list of the roles required to successfully conduct the business analysis process is included in the business analysis plan. Key process decisions are also included, for example, the approach for prioritizing, documenting, validating, communicating, approving, and changing requirements.

All planning decisions should be documented in a straightforward and clear manner so that stakeholders know what to expect when business analysis activities begin. Where team members disagree with one or more of the decisions being made, the business analyst assumes responsibility for negotiating and bringing the team to consensus. Once decisions are made, these should be documented so conflicts do not resurface later when the business analysis work is being performed.

The business analysis plan should be written in a manner that will be easily understood, because it will be reviewed and may need to be approved by key stakeholders. When developing the business analysis plan, it is a good practice to provide explanations for the planning choices made. For example, for projects using an adaptive life cycle, the depth and cadence of analysis activities will be planned much differently than for projects using a waterfall approach. The prioritization process, types of techniques, and deliverables will vary. Explaining why planning choices were selected provides context for those who review the plan and provides the rationale for the decisions made.

Generally, decisions that are reflected in the business analysis plan and are influenced by the selected project life cycle include:

- Type of elicitation activities to be conducted;
- Requirements analysis models that are required or expected;
- How requirements will be documented and communicated to stakeholders, including the use of any specialized tools;

- Business analysis deliverables to be produced;
- Roles and responsibilities for those participating in the requirement activities;
- How requirements will be prioritized, approved, and maintained;
- List of requirement states that will be tracked and managed in the project;
- How requirements will be validated and verified; and
- How the acceptance criteria will be determined for the requirements and solution validation.

3.4.2.1 Determining the Proper Level of Detail

The business analyst should ensure that a sufficient level of planning occurs in order to minimize the risks of developing poor estimates; misguiding stakeholders regarding the amount of required commitment; and miscommunicating the important decisions regarding requirement signoff, prioritization, and change management. The business analyst should avoid being too prescriptive and try to find a balance in the amount of planning performed. When rolling wave planning is used, consider the planning horizon and provide planning details only for the activities that are on the short-term horizon.

- **Balancing between flexibility and management.** The type of information commonly contained in a business analysis plan varies because each organization develops the template a bit differently depending upon organizational needs. When developing a business analysis plan, the business analyst should not attempt to document every aspect of the business analysis approach as that tactic is time-consuming and ineffective.
- **Do not sacrifice good management practices for flexibility.** The business analysis plan is intended to provide enough guidance to ensure quality in the business analysis process and is not intended to constrain or remove flexibility when an adaptive approach is needed to perform effectively. The plan should focus on documenting the key decisions about how the business analysis process will be performed and should not try to plan out each and every step of the process.

3.4.3 Understand the Project Context

There is no single approach to the business analysis effort that works best for every project, because all projects are unique. It is essential to understand the characteristics of a project and the context in which it is being conducted in order to define an appropriate approach. A project that is highly complex because it involves several interfaces or impacts a large number of business units requires a different approach to business analysis work compared to a project that is smaller in size and less complex.

The business analyst conducts some initial analysis including a review of the business case, project goals, and objectives in order to obtain the necessary context required to start defining the approach. Before planning decisions are made, the business analyst analyzes the characteristics of the project in order to determine the approach, including but not limited to the following:

- Project size,
- Project complexity,

- Type of project,
- Risk level,
- Risk tolerance,
- Geographic distribution of stakeholders,
- Aggressiveness of project timelines,
- Selected project life cycle,
- Imposed constraints and/or regulations,
- Market conditions and/or competitive landscape,
- Technology trends,
- Level of detail and formality required for deliverables,
- Time to market requirements, and
- Experience level of the business analysts assigned to the project.

There may be mandatory organizational standards that impose prescribed methods for conducting the work or require business analysts to follow a predefined process for defining, tracing, approving, or managing requirements. An organization may also influence the approach by requiring the use of standardized requirements templates. The business analyst needs to consider all of these factors when planning how to perform the business analysis work for the project.

3.4.4 Understand How the Project Life Cycle Influences Planning Decisions

A project life cycle is the series of phases that a project passes through from its initiation to its closure. The life cycle provides the structure for managing the project and is determined by factors such as managerial preferences, project characteristics, or how the organization desires to maintain and control projects. A number of process and planning decisions are dependent on the chosen project life cycle; therefore, the business analyst should tailor process decisions in accordance with the project life cycle selected.

Project life cycle models range from predictive (fully plan-driven) to adaptive (change-driven), and hybrid approaches fall anywhere between the two. The following is a high-level comparison of the three main structure types:

- Predictive:
  - Emphasis is on minimizing uncertainty.
  - Scope is entirely defined up-front.
  - Time and cost estimates are determined for the entire project when scope is defined.
  - Business analysis is conducted mostly up-front; requirements are completed before product development begins.
  - Deliverables are determined at the onset of the project.

  - Changes to scope are carefully managed.
  - Business value is delivered through one implementation.
  - The need and solution are known and do not change throughout the project.
  - Predictive life cycle methods are also referred to as plan-driven, traditional, or waterfall methods.
- Iterative/Incremental:
  - Project is split into phases and project phases are intentionally repeated.
  - Project work is performed sequentially or with some overlap in iterations.
  - High-level scope is defined up-front and the detailed scope is elaborated upon for each iteration.
  - Scope for future phases is defined when the prior phase starts or completes.
  - Product is developed iteratively as features are added incrementally.
  - Business analysis is performed up-front and then in small amounts throughout the project.
  - Requirements analysis is performed in time-boxed iterations.
  - The need and solution become more stable as the project progresses.
- Adaptive:
  - Business value is emphasized over minimizing uncertainty.
  - Time and cost estimates are fixed for each iteration.
  - Iterations are conducted quickly.
  - Overall scope is agreed to early. Detailed scope and product requirements are defined for a single iteration at a time.
  - Changes are expected; when new requirements are presented, these are captured in a product backlog, and then the backlog is reprioritized.
  - Business value is delivered iteratively.
  - Business analysis is constant.
  - The need and solution are unknown and unstable.
  - Adaptive life cycle methods are also referred to as change-driven or agile methods.

The project life cycle impacts a number of decisions about the business analysis process, such as:

- Business analysis activities that are to be performed,
- Order in which the activities will be completed,
- Timing of activities,
- Deliverables,
- Level of formality required for deliverables,
- Approach for prioritizing requirements, and
- Methods for addressing requirement changes.

3.4.5 Ensure the Team is Trained on the Project Life Cycle

Project teams and key stakeholders may require training if they have not worked previously with a selected project life cycle. The business analyst should be aware of the experience levels of the stakeholders and identify areas where coaching or training may need to be provided to assist the stakeholder through the requirement activities. Because templates and business analysis deliverables vary based on the project life cycle, time should be provided to update stakeholders who have the responsibility for reviewing, prioritizing, and validating requirements. In addition, the business analyst lead who manages a team of business analysts should ensure that the business analysts have the necessary skills to use the techniques and tools selected for the work.

> ***Collaboration Point***—Business analysts are responsible for stakeholder engagement in the business analysis process and the project manager is responsible for stakeholder engagement across the project. There is an opportunity for the roles to work together to determine the readiness and willingness of key stakeholders to participate in project activities.

3.4.6 Leverage Past Experiences When Planning

3.4.6.1 Lessons Learned

Lessons learned is the knowledge gained during a project, which shows how project events were addressed or should be addressed for the purpose of improving future performance. The lessons learned process is a best practice used by project teams to discuss, analyze, and document feedback about completed project activities. Lessons learned sessions are typically scheduled after the conclusion of a major phase or completion of a project. These sessions may be held more frequently as the need arises or when major events that require attention occur.

When determining a business analysis approach, the business analyst looks for similarities with other ongoing projects or previous projects that had successful business analysis approaches, reviews lessons learned documentation, and applies relevant lessons in whole or in part to the current business analysis process. For example, the business analyst may look through a previous business analysis plan to pull out estimating metrics and find ways in which the estimates can be improved for the current business analysis process.

> ***Collaboration Point***—Project managers and business analysts may each conduct lessons learned sessions; the project manager obtains feedback on completed project activities and the business analyst focuses on completed business analysis work. Instead of working independently, the roles should collaborate and facilitate the session together.

3.4.6.2 Retrospectives

In adaptive life cycle projects, retrospectives are meetings that are scheduled on a regular basis or conducted when a body of work is completed, such as the conclusion of an iteration or at the end of a project phase. Retrospectives have been used for many years in the development process of Kanban, agile approaches (e.g., Scrum

and eXtreme programming), and in Lean methods, such as Kaizen and continuous improvement. The purpose of a retrospective is to task the project team with identifying those areas where team performance can be improved. The team reflects on their successes and areas for improvement. The facilitator takes responsibility to ensure everyone in the room participates and works together to determine a course of action. Typically retrospectives use the following steps:

- **Set the stage.** The context and tone for the meeting is established.
- **Gather data.** Factual and relevant data is pulled together to support feedback.
- **Generate insights.** The team looks for commonality in the feedback including cause and effect.
- **Decide what to do.** The team collaborates to determine a course of action for improvement.
- **Close the Retrospective.** The session is ended.

This process is often facilitated with a list of simple questions:

- What is working?
- What is not working?
- What needs to change?

The facilitator uses charts and other visual methods to capture the information presented. The process is highly collaborative.

The biggest differences between lessons learned and retrospectives are in the timing and speed by which issues raised are addressed and the formality around documenting the learnings. Because adaptive life cycles offer more opportunity, retrospectives occur regularly and frequently, such as at the end of each sprint, or at the end of a Kanban delivery. Retrospectives may also be scheduled on a once-a-week basis, irrespective of delivery. Retrospectives are conducted in a highly collaborative fashion and decisions made are most often implemented with little formal documentation. Lessons learned are conducted at the end of stage gates or a phase such as a project closeout or when events occur that are worth learning from. Although lessons learned can be conducted more frequently, these are used less frequently than retrospectives and may be driven by the occurrence of an event versus a fixed schedule. Learnings discussed are formally documented and stored in a repository for future reference. Project teams leverage the lessons learned repository as an input prior to planning work on subsequent projects.

Due to the frequency by which retrospectives occur and because they are held multiple times over the course of a single project, improvements can be applied immediately, allowing the value to be gained from the proposed improvement in the next iteration.

When lessons learned are performed within a predictive life cycle, the learning can be applied on the same project or future projects. When improvements are proposed on future projects, the project context or characteristics may be different; therefore, identifying applicable lessons that add value and address performance can prove to be challenging.

To ensure that the frequency of retrospectives does not diminish their value, it is important that retrospectives are kept fresh to avoid mundane gatherings where time is not well spent. The team needs to meet often enough

to conduct valuable sessions but not too often that the task is considered a waste of time. Retrospectives assist in generating ideas to improve team performance but are not a replacement for training and coaching staff on process. Retrospective and lessons learned data are valuable inputs when formulating an approach for how to conduct the business analysis work for a project. The information gained from both can be leveraged to ensure process mistakes are not repeated and that valuable steps are continued.

3.4.7 Plan for Elicitation

At this stage of planning, the business analyst should begin to think about how and when to elicit, which techniques to use, and the sequence of the elicitation activities. Although it is appropriate to start thinking about technique options in planning, it is not important to prescribe each and every technique that will be used throughout the process.

Planning is iterative and much of the information will not be known or uncovered until the elicitation work is underway. When decisions or assumptions impacting the elicitation work are known and worthy of noting (e.g., decisions or assumptions that provide clarity or identify dependencies) the business analyst should note this information.

There are a number of factors to consider when planning for elicitation:

- Stakeholder group characteristics and dynamics,
- Project life cycle,
- Characteristics of the technique (e.g., speed of use, particular advantages),
- Type of project,
- Time constraints,
- Budget,
- Number of stakeholders,
- Location of stakeholders,
- Types of requirement deliverables being produced,
- Level of detail required in business analysis deliverables, and
- Techniques that are familiar to the business analyst and key stakeholders involved.

The business analyst should be aware that elicitation consumes project resources; therefore, there is a cost for each facilitated workshop completed, each interview conducted, and every survey sent. The business analyst balances the needs of the project against the quantity of requirements elicitation conducted with the goal of eliciting the right amount to sufficiently define the product solution.

Elicitation planning helps to ensure sufficient time is allocated for the elicitation work. It is easy to misjudge the level of effort required. For example, when a survey comprises a part of the elicitation effort, sufficient time needs to be allocated to design, build, administer, and analyze the survey results. When subsequent work is dependent on the survey results, the business analyst should identify and plan for that also.

When insufficient time is allocated for elicitation activities, dependent downstream tasks may be delayed when business activities extend past the projected timelines. Key stakeholders may find it difficult to allocate additional time to participate in unplanned elicitation sessions when these sessions are planned at the last minute. The need to accommodate additional, unexpected workshops begins to have a multitude of impacts to the project schedule. It is best to take a judicious approach to planning up-front but to ensure enough planning is performed so that dependencies can be identified and relied upon to schedule downstream activities.

3.4.7.1 Strategies for Sequencing Elicitation Activities

Where to start eliciting requirements often poses a challenge. Business analysts know they need to acquire a lot of information, they are constrained by the project timeline, their stakeholders have a limited number of hours to dedicate to the project, and their stakeholders have business knowledge and expertise in specific areas. All of these factors need to be considered to determine the optimal sequence for conducting elicitation activities.

There are a number of strategies the business analyst may consider when determining which areas of requirements elicitation to address first. Suggested areas of focus are where there are:

- Significant amounts of business value to be gained,
- Greater risks,
- Many project unknowns or uncertainties,
- Significant technical challenges,
- Dependencies on other components or interfaces,
- Required third-party resources that the project is dependent on, and
- Limited number of resources or a risk that a key resource may leave the project or company.

Other constraints that affect elicitation sequencing are the dates key stakeholders impose on the project. For example, plant shutdowns or seasonal constraints where stakeholders are committed to the operational work of the business before the work of the project.

> ***Collaboration Point***—When deciding the order of business analysis activities, the project manager and business analyst should work together to determine how the resource availability will impact sequencing decisions.

3.4.8 Plan for Analysis

Planning for analysis occurs at a high level in much the same manner as elicitation planning. The business analyst will begin to think about which analysis techniques will be best suited to meet stakeholder and project needs. Although it is not necessary to call out each and every analysis technique that may be used, the business analyst should start to think about how the analysis work may be performed and determine the tools, templates, or training that need to be acquired prior to the start of the actual work.

There are a number of analysis techniques to choose from, and each technique has its strengths and circumstances where each is best used. At the planning stage, the business analyst does not have sufficient information to know all of the analysis techniques that will be used, since it is unknown as to what type of information the elicitation activities will produce. At the planning stage, the business analyst is just starting to think about how to approach the analysis activities and much of what is considered here may change as the elicitation activities proceed. Planning for analysis is iterative and occurs throughout the project.

3.4.9 Define the Requirements Prioritization Process

Prioritizing requirements is an important step in managing product scope. There are many ways to prioritize requirements; therefore, the business analyst determines which approaches are going to be used on the current project. To minimize conflict during the requirements prioritization activities, the business analyst should define the process for establishing priorities before the requirements are elicited. Setting expectations early with stakeholders helps to minimize situations where stakeholders become unhappy when their requirements are prioritized to the bottom of the list. Business analysts will find themselves leveraging their negotiation and conflict management skills heavily during prioritization discussions.

When building the business analysis plan, the business analyst should include an explanation as to how prioritization will be conducted for the project. Some common factors to consider when defining the prioritization process are:

- When requirements prioritization will occur,
- The likelihood that priorities will change,
- Stakeholders who will be involved in the prioritization process,
- Techniques that will be used for prioritization,
- Criteria that will be used to prioritize, and
- Stakeholders who will approve the prioritization decisions.

The project life cycle influences the prioritization process and often dictates the frequency, timing, and techniques for performing prioritization. For example, adaptive life cycles use prioritization techniques for each iteration in order to determine the features to be provided in the next release of the product. Projects using a predictive or waterfall life cycle will conduct prioritization up-front before product construction begins. Priorities may still shift in a waterfall project, but incorporating such changes is more difficult than in a project following an iterative development cycle.

In addition to defining the timing and frequency of prioritization, the business analyst should also define the criteria that will be used to prioritize. Requirements are prioritized based on a number of factors such as:

- **Value.** Addressing high-value requirements first to reap the financial or goodwill benefits up-front.
- **Cost.** Evaluating requirements based on financial costs or opportunity costs.
- **Difficulty.** Considering how difficult the requirement is to fulfill.

- **Regulatory.** Addressing regulatory or legislative requirements that have imminent compliance deadlines first.
- **Risk.** Implementing high-risk requirements first to uncover issues early.

Some common techniques for determining priority are MoSCoW, multivoting, timeboxing, and weighted ranking. For information about conducting the prioritization process, see Section 4 on Requirements Elicitation and Analysis where MoSCow, multivoting, and timeboxing are discussed. See Section 2 on Needs Assessment for more information about the weighted-ranking matrix and weighted-ranking approach.

3.4.10 Define the Traceability Approach

Traceability involves documenting requirement relationships and is used in business analysis to maintain product scope. When a traceability approach is defined and adhered to, requirement relationships are created, which allow the project team to trace backwards to identify the origin of a requirement, trace forward to identify how the requirement was tested and implemented, or trace in-between requirements to provide insight into the value a group of related requirements can deliver. Requirements that are documented but fail to trace to a business need are considered out of scope. Requirements that fail to trace to a solution component identify areas where the product is not in compliance with the requirements.

When sufficient traceability is established, it is much easier for the project team to understand how a proposed change will impact the project. Those requirements affected by a change can be evaluated to understand how a change in that requirement impacts (a) the related project components already built to satisfy the requirement or (b) test cases already created to test the feature, etc. A sufficient amount of traceability ensures that the impacts of requirements change are properly assessed and quantified from a risk, cost, and time perspective.

During planning, the business analyst should determine what level of traceability is going to be performed on the project. Higher-risk or more complex projects may require more traceability. Understanding the project context, the type of project, and any organizational process or compliance standards that are required, the business analyst then documents the traceability approach in the business analysis plan.

The types of traceability decisions the business analyst should consider are:

- Types of requirements to be traced,
- Level of detail to trace to,
- Relationships that will be established and maintained,
- Requirement attributes to be tracked,
- Requirement states that drive the requirements life cycle (for example, approve, defer, reject, etc.),
- Tools used to perform the traceability, and
- Process decisions regarding how traceability will be established and maintained.

As the size and complexity of the project increases, so does the complexity of the traceability. Larger, more complex projects increase the number of relationships that need to be established between the components. As the amount of traceability increases, the time and cost to manage the established linkages increases as well.

To ensure the traceability approach is adequately sized for the project, the value of the process should be compared against the time and cost required to establish and support it. Risks should also be analyzed to ensure additional risks are not introduced by proposing a traceability process that increases the likelihood of missing requirements in the final product.

> ***Collaboration Point***—The right amount of traceability helps to reduce project risks, but too much traceability works against the project team and incurs costs and schedule impacts unnecessarily. The project team has an opportunity to collaborate and define a traceability approach that is suitable for the needs of the project. Business analysts should not define the traceability approach independently as decisions on traceability are very interrelated to project factors that the project manager is responsible for managing.

For further information on establishing and maintaining traceability, see Section 5 on Traceability and Monitoring.

3.4.11 Define the Communication Approach

A communication approach describes how business analysis information will be structured and when communication to stakeholders will occur. The approach can be formally documented in the business analysis plan or in a separate business analysis communication plan if the project or organizational needs require this level of structure or formality.

The business analyst should discuss the communication process with stakeholders to ensure the approach will meet expectations.

When defining the business analysis communication approach, consider the following:

- Types of requirements and requirements-related information that will be communicated,
- Who requires communications and what information they expect,
- Stakeholder preferences for receiving information (e.g., summary level or detail),
- Preferred delivery methods for distributing or accessing requirements and requirement related information,
- Level of formality required in the communication, and
- Tools, including requirement repositories, which stakeholders will need access to.

In addition to gaining agreement on what information will be communicated, the business analyst also receives input about how stakeholders wish to receive information on requirements and how often they wish to receive information. For example, upper level management may request access only to requirements once the requirements are approved. Subject matter experts may request access to the requirements throughout the project. It may not be appropriate to copy every stakeholder on each communication; therefore, the business analyst may request the business to identify a key stakeholder representative who can serve as the recipient of the requirement-related communications.

The time zone and physical locations of the stakeholder should be considered when defining the communication approach. Location may impose constraints on the communication process as well as the tools selected to support

the process. When stakeholders span across multiple time zones, which makes it difficult to have direct contact with the stakeholder, then the communication process may need to be formalized to ensure information is delivered in a clear and concise manner and can be easily understood when the practitioner is unavailable.

> ***Collaboration Point***—The business analyst should not create the communication plan independently of the project manager. If the business analysis communication plan is formally documented, agreement should be reached on whether the information will be kept as a subplan or included within the project manager's communication plan.

3.4.12 Define the Decision-Making Process

Throughout the business analysis process, there are a number of decisions that need to be made before work can commence. The decision-making process should be defined collaboratively with stakeholder input. Once the team determines the decision process, it is documented within the business analysis plan so the team has a point of reference when the work begins. When there are differences regarding how decisions should be made, the business analyst facilitates agreement and clears up conflicting viewpoints prior to the execution of the business analysis activity. A thorough understanding of the types of decisions that stakeholders will be required to make and an assessment of the authority levels in the process will save tremendous time and confusion later on when decisions need to be reached by the requirements team.

The following information can be considered when defining the decision-making process:

- Types of decisions that will be made, including how requirements will be approved,
- Roles and authority levels, for example, who is involved in the discussions and who makes decisions, etc.,
- Process to follow when consensus cannot be reached and requirements-related issues need to be escalated,
- Required turn-around time for a decision to be reached,
- How decisions are documented and communicated, including requirements signoff, and
- Special tools or techniques that may be used to help teams evaluate alternatives, for example, decision analysis, decision modeling, decision trees etc.

3.4.13 Define the Requirements Verification and Validation Processes

Two business analysis activities that often confuse stakeholders are requirements verification and requirements validation. According to the *PMBOK® Guide* – Fifth Edition, verification is the evaluation of whether or not a product, service, or system complies with a regulation, requirement, specification, or imposed condition. Validation is the assurance that a product, service, or system meets the needs of the customer and other identified stakeholders. Requirements and associated requirement models are both verified and validated.

In business analysis, requirements verification is the process of reviewing requirements and models to ensure they meet quality standards. Verification is performed to ensure that requirements are constructed properly and

that models are clear enough to be used effectively. Verification activities ensure that the right product will be built. Verification can be conducted iteratively as requirements are in development. All requirement types should be verified, for example, business requirements, stakeholder requirements, solution requirements, etc.

Requirements validation is the process of ensuring that all requirements accurately reflect the intent of the stakeholder and that each requirement aligns to one or more business requirements. Validation is performed through the use of structured walkthroughs and traceability. Conducting a structured walkthrough provides an opportunity for the business analyst and stakeholders to collaborate and to ensure the requirement as stated is correct. Requirements validation ensures that the requirements accurately reflect what the project is expected to deliver. Validation ensures that the right product is being built.

During business analysis planning, the business analyst defines how verification and validation activities will be conducted. Stakeholders often have expectations about how they want to review requirements and, therefore, their input should be taken into consideration when defining these processes. Organizations may already have documented standards that identify the characteristics for good requirements. Quality standards should be defined in planning so they are known prior to requirements development and verification activities. Verification processes often leverage checklists to define the quality attributes. When a checklist will be leveraged for the verification process, the business analyst ensures that the checklist is created before the verification activities begin. Requirements validation and verification and the characteristics for quality requirements are further discussed in Chapter 4 on Requirements Elicitation and Analysis.

3.4.14 Define the Requirements Change Process

The requirements change process is performed differently depending on the selected project life cycle. For a project following a predictive life cycle, the project team may use a formal change control process; whereas a project following an adaptive life cycle may not. Adaptive approaches expect that requirements will evolve over time and, as a result, often take a flexible approach to requirements change control.

For projects that use a formal change control process, the business analyst should consider documenting the approach so that stakeholders know how to request a change and how changes will be assessed for impact. Changes to requirements may involve a change in scope, a restated business objective, or the addition or deletion of features to the product or service being developed; therefore, the business analyst will need to determine how requirements documentation should be updated once a change is approved.

Consider the following type of information when defining a requirements change control process:

- How requirement changes will be proposed,
- How changes will be reviewed,
- How change management decisions will be documented,
- How requirement changes will be communicated, and
- How changes to requirements, models, traceability, and other requirements-related information will be completed and made available once a change is approved.

The business analyst also needs to ensure the project team understands the roles and responsibilities for the requirements change process, for example:

- Responsibility for proposing changes,
- Responsibility for conducting the assessment or impact analysis,
- Roles that participate in change discussions, and
- Responsibility for approving changes.

When there is an approved change request form or impact analysis template that should be used, this information should be included when defining the requirements change process. For information on conducting the change control process, see Section 5 on Traceability and Monitoring.

> ***Collaboration Point***—The requirements change control process may be documented by the project manager in the change management plan. When there is a need to formally document the process, the business analyst and project manager should work together to determine who takes ownership for documenting the plan and whether the business analysis plan or change management plan is the source for this information.

3.4.15 Define the Solution Evaluation Process

Defining the solution evaluation process identifies how a solution is evaluated during the evaluation phase, which activities will be performed, what techniques will be applied, and how information will be analyzed.

Through evaluation, the business analyst determines whether a solution has achieved the desired business result. Evaluation is performed before and after a solution is implemented. When evaluation is conducted after a solution is implemented, new or changed requirements may be identified, which can lead to refinement of the solution or can generate new projects for the purpose of creating a new version of the product. The project life cycle influences the timing and frequency of the evaluation activities.

The expected benefits for the product or solution were defined in the business case. The work in planning is to determine how to measure these benefits.

Solution evaluation planning involves defining the following:

- Evaluation criteria and acceptance levels;
- Qualitative and quantitative evaluation activities to be performed;
- How the solution will be evaluated;
- When and how often evaluation will be performed;
- Evaluation techniques that will be used, for example, focus groups, observations, surveys, etc.;
- Whether any special measurement tools will be required; and
- How results will be analyzed and reported.

For more information on conducting the evaluation process, see Section 6 on Solution Evaluation.

3.5 Plan the Business Analysis Work

3.5.1 Determine Who Plans the Business Analysis Effort

One point of confusion on projects is the overlap that exists between business analysis planning and project management planning. The confusion over who performs what work is compounded by the fact that the responsibility for business analysis planning differs across organizations, by type of project and by project methodology. Even within a single organization, the business analyst may be assigned to build the work breakdown structure for one project in order to define the business analysis work; and for a second project, the project manager may interview the business analyst to obtain the needed inputs for developing that plan.

The business analysis profession is progressing, and the role has become more defined over the years. Hybrid roles, for example the project manager/business analyst (PM/BA) positions have existed for many years and still exist in a number of organizations. As organizations have purposely split apart the responsibilities into two distinct roles, the practitioners who migrated into pure business analyst positions bring with them experience in project planning. When a business analyst is adept at planning, a project manager may leverage that experience and request assistance from the business analyst. It is not uncommon for a business analyst on a project with a predictive life cycle to draft a work breakdown structure to conduct initial conversations with the project manager regarding the business analysis work for the project.

> ***Collaboration Point***—Planning is an area where project managers and business analysts will find that their roles overlap. Ultimately the project manager is responsible for the project management plan and is accountable for managing all project activities. The business analyst can help support the project manager, because the business analyst has expertise with the business analysis process and can provide specifics on the business analysis work that is recommended to meet project objectives. Other team members should be consulted and included in plan development to ensure their buy-in and engagement in the work.

3.5.2 Build the Business Analysis Work Plan

The business analysis work plan is the project plan that covers the work to be performed during the business analysis process. The business analyst is responsible for identifying this work using all the project and environment information known to them, for example, selected project life cycle, project size, known risks, organizational process assets, etc. It is critical that the project manager be included and involved as the business analyst lays out this work, because ultimately the business analysis work plan will be integrated within the project management plan. Any discrepancies or misalignment needs to be resolved before work commences.

The first step in building the plan is identifying the deliverables that the business analyst is responsible for producing.

3.5.2.1 Identify the Deliverables

The business analyst collects many forms of information over the course of the project, but not all information collected and documented is considered to be a project deliverable. According to the *PMBOK® Guide* – Fifth Edition,

a deliverable is any unique and verifiable product, result, or capability to perform a service that is required to be produced to complete a process, phase, or project.

Agendas, meeting minutes, parking lot lists, and some types of models, although necessary to organize work and perform effectively, are not required to be produced and, therefore, are not considered a business analysis deliverable. These items are often referred to as work products and are created in order to perform the work, but are not a promised deliverable that will be tracked and managed through the project management plan. During planning, the business analyst should think about the work products and how they will be created, stored, accessed, and used by the project team. Planning for work products is much less formal than deliverables.

Deliverables, on the other hand, are required and often impact the work of individuals who perform activities later on in the project life cycle. These documents are usually formatted for consistency across projects and often have an approved structure or template that the business analyst is required to use. Examples of these include, but are not limited to: a high-level requirements document, software requirements specification, user stories, business rules catalog, and a nonfunctional requirements list. While planning the deliverables for the project, the business analyst considers what types of requirements will be collected and the different deliverable formats that are required to be used. Additionally, consideration is given to stakeholder requests for receiving specific types of documentation or obtaining information in certain formats or tools. The selected project life cycle heavily influences which deliverables will be produced as do the size and complexity of the project.

The business analyst should consider the following when planning the business analysis documentation:

- What deliverables will be produced;
- How deliverables will be used by other team members and key stakeholders;
- How changes to deliverables will be managed;
- Required formality of the deliverables;
- How deliverables will be accessed and by whom;
- Tools that are required to produce, maintain, or store deliverables; and
- Whether requirements will be reused on future projects.

  Collaboration Point—The business analysis deliverables produced for the project are determined by the business analyst but these decisions are not made independently. The business analyst solicits input from the sponsor, project manager, and key stakeholders to ensure that expectations are met. The business analyst should work closely with the project manager, because deliverables impact the project schedule and can create dependencies for other downstream project work.

3.5.2.2 Determine the Tasks and Activities

As the stakeholder and project characteristics become better understood during the planning process, the business analyst considers the business analysis work that needs to be performed in order to conduct an effective business analysis process. After agreeing on the list of deliverables that will be produced to satisfy the project objectives, the business analyst begins to define the activities required to produce them. Each deliverable requires

the completion of one or more activities. The project team should work together to define the task list. Task identification is an iterative process. The business analyst may use a number of techniques to help identify a complete list. Brainstorming, decomposition modeling, interviews, and lessons learned are a few of the techniques that can be used.

- **Decomposition model.** A decomposition model is an analysis model used to break down a high-level object into smaller more discrete parts. Typical objects often analyzed with decomposition may be solution scope, organizational units, work products, processes, functions, or any other object types that can be subdivided into smaller elements. A decomposition model is easy to produce and the models are easily constructed and understood by stakeholders. Decomposition models provide a way to analyze highly complex concepts, because these allow for the complexity to be broken into manageable elements. The hierarchical relationships are easily portrayed through the levels.

  Decomposition models cannot be used to show sequence and process steps. In business analysis planning, a decomposition model is used to identify business analysis tasks, activities, and deliverables by detailing out the business analysis work. When conducting stakeholder analysis, a decomposition model is used to analyze the organization with the goal of discovering stakeholder groups. On IT projects, this type of model is helpful to break down solutions into solution components to further understand their features. Decomposition models are also referred to as decomposition diagrams.

3.5.2.3 Determine the Timing and Sequencing of Tasks

The timing of business analysis tasks is directly influenced by the selected project life cycle. Sequencing is also influenced by a natural order, because some tasks require the completion of outputs produced by other activities. The business analyst analyzes these factors and may consider one or more of the following when determining the sequence of activities for the work plan:

- Availability of resources required to perform or contribute to the work,
- Needs of downstream recipients whose work relies on the business analysis deliverables,
- Relationship of project work to other work being performed within the organization,
- Contractual and statement of work obligations,
- Dependencies on training and tools to perform the business analysis work,
- Staffing and new hire needs, and
- Risk and complexity level of tasks.

3.5.2.4 Determine the Roles and Responsibilities

Project managers often develop a RACI to identify roles and responsibilities for the resources assigned to the project. RACI analysis is also performed in business analysis when determining roles and responsibilities for the business analysis effort. The business analyst cannot assume that everyone involved in the business analysis process knows their role or what they are accountable for. Key stakeholders may be confused when they participate on more than one project team at a time. Stakeholders may have confusion about the roles that are assigned to

them versus the roles they desire. While the RACI is used to document the role responsibilities across the team, it is also valuable to use this technique to define the role responsibilities between the project manager and the business analyst. Determining roles and responsibilities at the start of the project helps to minimize confusion and conflicts, especially in areas where responsibilities appear to overlap.

In business analysis activities, it is essential to understand who is responsible for providing requirements, who has authority to approve requirements, who can make decisions or settle differences when they arise, and who is involved when changes to requirements are proposed. The RACI technique can be used to facilitate, sort through, document, and gain approval on these decisions. In addition, the RACI can be used as a communication tool. It can be referenced throughout the business analysis work when conflicts arise over authority and responsibility levels. Once completed, the RACI matrix can be included as part of the business analysis plan to ensure these decisions are approved and managed. How to construct a RACI is described in Section 2 on Needs Assessment.

> ***Collaboration Point***—The business analyst and project manager share a common interest in ensuring that key stakeholders fully understand and agree to the roles and responsibilities they hold on a project. While the project manager is managing these expectations across the project, the business analyst is managing these expectations during the business analysis activities. Both roles have an opportunity to work together to help ensure the expectations regarding stakeholder involvement are met.

3.5.2.5 Identifying the Resources

The business analyst will review the planned work, determine which resources will be needed for the work, and identify any areas where resources may be overallocated or missing. As a minimum, the business analyst needs to ensure that all functional areas impacted by the project are represented by a resource during requirements elicitation, validation, and approval activities. As subject matter experts are often coveted across the organization and often allocated on multiple projects, business analysts should raise any concerns obtaining commitments from resources once the resources are assigned to the project.

> ***Collaboration Point***—The business analyst can lend their expertise and personal insight to the project manager as they work collaboratively to determine the optimal number of subject matter experts to engage on the business analysis activities. The business analyst may have prior experience working with key resources and specifics that influence task allocations and assignments that the project manager may benefit from prior to finalizing the overall resource plans for the project.

3.5.2.6 Estimate the Work

As the work plan details begin to emerge, the business analyst plays a key role in defining the level of effort required for the defined activities and deliverables. There are a number of estimation techniques that may be used to help support the estimation process. The project manager may have a preference as to which techniques are applied.

To help improve the reliability of the estimates, the business analyst may seek the support of peers within the organization who have more years of experience in estimating similar projects within the industry or line of business.

These resources may review proposed estimates and suggest adjustments based on their experience. Business analysts may consider consulting with the key stakeholders who will be involved in the work. Business analysts may validate estimates by reviewing the planning information from past projects that used similar elicitation and analysis techniques.

The following considerations should be factored into the estimates:

- Project size and complexity,
- Selected project life cycle,
- Amount of ambiguity surrounding the proposed solution (e.g., ambiguity will be higher when taking advantage of an emerging technology or using a solution that the organization has little experience with),
- Number of stakeholders and stakeholder groups involved in the elicitation activities,
- Types of elicitation techniques being considered,
- Location of those involved in the business analysis activities (including the product development and testing resources who the business analyst will interface with),
- Schedule and budget constraints,
- Any known assumptions,
- Number of business analysis deliverables being produced,
- Formality and level of detail required for the business analysis deliverables, and
- Experience level of the project team (including the business analysis team assigned to the project and the stakeholders participating in the requirements-related activities).

Estimation is an iterative process and estimates should always be revisited as the project progresses and more information becomes known to the project team. For an adaptive life cycle, the analysis work may not be estimated separately but may be included in an overall estimate of what it will take to deliver an increment of the solution.

> ***Collaboration Point***—The project manager and business analyst should determine how revisions to the business analysis estimates will occur. Project managers facilitate the estimation work when building the project management plan. Business analysts may be asked to facilitate the estimation work for the business analysis activities, since they are responsible for the business analysis process and more familiar with the specifics. Estimating the business analysis activities is a potential area of overlap between the two roles; therefore, the business analyst and project manager will benefit greatly by collaborating to determine who should complete and revise these estimates throughout the project.

3.5.3 Assemble the Business Analysis Work Plan

Once the business analysis deliverables, tasks and activities, and required resources for completing the work are known, the information is assembled in a plan that clearly depicts the timing of the work and any dependencies. This is a point of integration between business analysis and project management. The business analyst and project manager need to agree on the level of detail needed for the project management plan. The level of detail maintained

in the project management plan can vary based on a number of factors minimally depending upon:

- Complexity of the business analysis effort,
- Size of the project,
- Amount of business analysis work being tracked, and
- Type of project life cycle.

The business analysis work plan may be included within the business analysis plan along with the other planning decisions or organizational preferences, and the selected project life cycle can influence the approach used to document and communicate the work plan. See Table 3-1 for a sample business analysis work plan.

> ***Collaboration Point***—It is beneficial for the project manager and business analyst to collaborate on how to document and communicate the work plan to ensure that the business analysis activities and schedule properly integrate into the project management plan. When the business analysis process is highly complex, the project manager may prefer that the low-level details regarding the business analysis activities be tracked by the business analyst in a separate business analysis work plan.

Table 3-1. Sample Business Analysis Work Plan

Task Name	Resource	Level Of Effort
Use Case Diagrams	S. Bhomack	34 hours
Use Case Specifications		
Use Case 1		
Interview SMEs	A. Manach	10 hours
Draft Use Case	A. Manach	32 hours
Review Use Case	A. Manach	2 hours
Refine Use Case	A. Manach	5 hours
Approve Use Case	A. Manach	1 hour
Use Case 2		
Interview SMEs	E. Simone	12 hours
Draft Use Case	E. Simone	41 hours
Review Use Case	E. Simone	3 hours
Refine Use Case	E. Simone	6 hours
Approve Use Case	E. Simone	1 hour
Use Case 3		
Interview SMEs	S. Bhomack	7 hours
Draft Use Case	S. Bhomack	22 hours
Review Use Case	S. Bhomack	1.5 hours
Refine Use Case	unassigned	3.5 hours
Approve Use Case	unassigned	1 hour

3.5.4 Document the Rationale for the Business Analysis Approach

Documenting the basis for the decisions made during the planning efforts helps to reduce uncertainty. Sharing the rationale with the stakeholders provides assurance that the plan is thought out and purposeful. For the project team, understanding the rationale behind the chosen business analysis approach provides context and helps answer questions regarding why the work is being conducted and why it is being performed in a specific manner. For the project manager, understanding the rationale for the business analysis approach provides the context needed to support and fund the work, manage it, and justify the roles and resources required to complete the work. The sponsor should understand the approach to champion, lend support, and rally stakeholders around the importance of the business analysis activities.

The business analysis plan along with the documented rationale provides valuable information for future project teams. When business analysis work is being planned for a new project, reviewing past plans and lessons learned assists with the planning work and allows analysis decisions to be based on experience. A Center of Business Analysis Practice may help to formalize the collection and storage of these assets, but for organizations that do not operate a center, an informal process for storing this information in a shared repository serves the organizational need.

3.5.5 Review the Business Analysis Plan with Key Stakeholders

Once the business analysis plan is completed, the business analyst reviews the plan with the key project stakeholders. Reviews are conducted in person; however, when stakeholders are distributed in different geographical locations, a collaborative approach may be used that allows stakeholders to raise questions and receive direct feedback to their concerns.

The objective for the review is to reduce the risk of stakeholders failing to support business analysis activities when the work begins. Stakeholders need to be comfortable with their role in the process, be aware of the time commitment to participate in the process, and understand how decisions about requirement priorities and changes are handled. Conducting these discussions up-front helps to obtain buy-in early and minimizes the risk of stakeholder conflict when the activities are underway.

There should be sufficient stakeholder involvement during these reviews to ensure stakeholder needs are adequately addressed. When stakeholders feel a sense of ownership about the process, they are more likely to demonstrate a higher level of interest and remain more engaged in the business analysis activities. The review process is a way to reduce the possibility of overlooking a stakeholder characteristic, concern, or limitation that could negatively influence the business analysis process. Reviews can reveal areas where stakeholders are confused or unclear about the requirements-related activities. Reviews sometimes uncover unknown risks such as a shortage of resources, schedule conflicts, or concern over assigned roles.

Key stakeholders in the business analysis activities should be invited to participate in the review process, for example, those having:

- Responsibility for approving, prioritizing, or validating requirements;

- Responsibility for a role in the change control process;
- Management responsibility for staff participating in the requirement activities; and
- Responsibility for serving as a subject matter expert on the project.

Additionally the project sponsor, project manager, and lead person who represent the interests of the downstream recipients of the requirements, such as the software development manager, quality assurance manager, and training manager should be included.

> ***Collaboration Point***—The business analyst should ensure to discuss the business analysis process with those who are tasked with performing the work. Key stakeholders who have an opportunity to provide input to structure the process are more likely to support the requirements-related activities.

3.5.6 Obtain Approval of the Business Analysis Plan

The business analysis plan may need to be revised to accommodate the feedback received from stakeholders. All stakeholder concerns may not be actionable, especially when the request imposes a risk to the project or reduces the effectiveness or quality of the business analysis process. The business analyst works through the issues raised and facilitates resolutions in order to avoid discouraging or disengaging stakeholders.

Once all concerns are resolved, the business analyst takes steps to obtain approval on the plan. Obtaining approval on the plan helps to reduce the following project risks:

- Stakeholders do not support the business analysis process once it starts.
- Project team underestimates the amount of time that business analysis activities will take.
- Funding allocated to the requirements phase of the project is inadequate.
- Key stakeholders underestimate the level of involvement or participation necessary for the requirement activities.
- Key resources are unavailable to participate in requirement activities, when required.
- Stakeholder expectations with regard to how requirements are documented and communicated are missed.

Approval may be formal and require a signature or may be informal and only require verbal acceptance. There may be an organizational process that defines how this approval should be attained, but when a process does not exist, the practitioner or project team should define the process. It is recommended to obtain approval from the sponsor and key stakeholders, such as the subject matter experts who will be participating in the requirements activities or anyone who approves requirements or participates in change activities. Once the business analysis plan is approved, updates to the plan should be made in a controlled fashion.

When the project uses an adaptive project life cycle, the team may forego a formal review and approval process since, in this case, much of the planning would have occurred as a result of extensive team dialogue and collaboration.

4

REQUIREMENTS ELICITATION AND ANALYSIS

4.1 Purpose of this Section

Requirements elicitation and analysis is the iterative work to plan, prepare, and conduct the elicitation of information from stakeholders, to analyze and document the results of that work, and to eventually define a set of requirements in sufficient detail to enable the definition and selection of the preferred solution. Within requirements elicitation and analysis, business analysts use a number of elicitation techniques and apply a number of analysis models to support the elicitation and analysis activities.

4.2 What it Means to Elicit Information

The common approach to obtaining requirements information is through "elicitation." Requirements elicitation is the activity of drawing out information from stakeholders and other sources. In business analysis, it involves eliciting information about the causes of the business problem or the reasons for addressing a current opportunity, as well as the information that will eventually be used to derive a sufficient level of requirements to enable solution development and implementation.

4.2.1 Elicitation Is More than Requirements Collection or Gathering

The elicitation process is more than collecting or gathering requirements. The terminology "collecting" or "gathering" requirements implies that stakeholders already have requirements ready to be collected or gathered. Requirements are not ready and waiting in the stakeholders' minds or in documents within the business community. While stakeholders may not have actual requirements, they do often have wants and needs; however, they may not be able to express them clearly. They may know there is a problem, but may not know exactly what the problem is. They may want to take advantage of an opportunity, but not know how to get started.

Part of the business analyst's job is to help the stakeholders define the problem or opportunity and determine what should be done to address it. The elicitation process helps facilitate this work. Some stakeholders may not even know their needs. For example, a start-up organization exploring potential opportunities may not have any idea of what to ask for. In this case, the business analysis process supports the discovery practices by eliciting and validating unknown needs. Stakeholders have information that is valuable to the business analyst, but the information is not necessarily in the form of a requirement that is ready and waiting to be collected.

4.2.2 Importance of Eliciting Information

The results of elicitation are a core input for business analysis work. Information is not only elicited to generate requirements or answer questions from the solution team, but the information becomes the basis to effectively complete other business analyst tasks, such as:

- **Support executive decision making.** Provides supporting information to the executives who are making strategic decisions about the direction of the organization. Considering that all such strategic decisions are business based, the business analyst objectively obtains the information necessary for these decisions to be made.
- **Apply influence successfully.** Influences to enable the completion of work. Business analysts have no vested authority; therefore, they should use influence in order to get things done. Influence is more successful when it is backed with information that supports the goal.
- **Assist in negotiation or mediation.** Negotiates on behalf of the project team, management, or business. In all such negotiations, the business analyst first elicits information and understands the motivations and/or goals of all sides in the conflict and then negotiates based on principle or objective facts.
- **Resolve conflict.** Provides information necessary to resolve conflicts. Business analysts are often brought in to resolve conflict or to serve as direct mediators for the project manager or business manager. Conflict is generally resolved with information, because conflict in business is usually the result of misinformation or assumptions based on a lack of information.
- **Define problems.** Provides the information to identify the real problem. The business analyst provides this information by identifying and analyzing the business processes that need improvement over the course of the analysis.

Failing to elicit enough information can result in erroneous conclusions and an increased number of assumptions, but too much information may hinder the team's ability to move forward. The art of elicitation is to obtain enough information to be able to validate requirements through a process of learning in order to confirm that the project team is delivering the right solution.

4.3 Plan for Elicitation

Some planning for elicitation occurred during the construction of the business analysis plan. Because planning is iterative, other planning is performed closer to when the activities are ready to begin. Prior to elicitation, the business analyst begins to focus on more specific details such as how to conduct elicitation, which stakeholders to involve, and which order to schedule elicitation sessions.

A well thought out approach to elicitation provides the following benefits:

- Clearer idea of the necessary information to define a problem, affect an improvement, or produce a solution;
- Fewer unnecessary elicitation activities;
- More valuable results from each elicitation session;

- More efficient and predictable use of stakeholder time to elicit the information; and
- Better overall focus on the entire elicitation process.

4.3.1 Develop the Elicitation Plan

An elicitation plan is a device used by business analysts to help formulate ideas about how to structure the elicitation activities. The plan is not a formal document and does not take a lot of time to create. It can be documented or can be the thought process used to prepare for the forthcoming elicitation efforts.

The work needed to create the elicitation plan involves thinking through how to best coordinate and conduct elicitation across a project. Some of the elements in an elicitation plan include but are not limited to:

- **What information to elicit.** What does the business analyst need to know in order to define the problem, solve the problem, or answer the question?
- **Where to find that information.** Where is that information located: in what document, from what source, in whose mind? Identify who has the information or where it is located.
- **How to obtain the information.** What method will be used to acquire the information from the source?
- **Sequencing the Elicitation Activities.** In what order should the elicitation activities be sequenced to acquire the needed information?

4.3.1.1 Finding Information

Sources of information may be (a) an individual the business analyst plans to elicit information from, (b) a model, or (c) other documented reference. Individuals in the elicitation plan should be identified by role rather than name. When information cannot be provided by individuals, the business analyst should consider leveraging basic elements of the enterprise architecture when one exists. Components within the enterprise architecture often contain reusable artifacts such as process flows, logical data models, business rule models, business domain models, etc. It is a good practice to try to identify at least two sources for each topic or question to be explored during elicitation, in order to avoid proposing any requirement or solution that is based on the opinion or information derived from a single source.

4.3.1.2 Techniques for Eliciting Information

There are several methods that are used to elicit information. Each technique has its own strengths and weaknesses. By understanding a variety of techniques, the business analyst has sufficient information to know which technique is best to use. Formal elicitation methods are planned and structured while informal methods are typically unplanned and/or unstructured. See Section 4.5.5 for a listing of commonly used elicitation techniques.

4.3.1.3 Sequencing the Elicitation Activities

There are dependencies that constrain the timing of elicitation activities. Some information is necessary before other information can be understood. Planning helps to identify and sequence information in the order the

information needs to be collected. These dependencies may be documented in the elicitation plan to help organize the flow of elicitation activities.

Table 4-1 provides an example of a completed elicitation plan to move office staff to a new location.

Table 4-1. Example of Completed Elicitation Plan

What Information	Source	Method	Sequence
How many employees will be moved?	HR	Interview	2
What is the physical layout of both buildings?	Building plans Facilities	Document analysis Interview	1
What equipment will be moved?	IT Facilities Executive VP	Meeting	4
Shall we move in all at once or in a phased approach?	HR Executive VP	Interviews	3
Is it possible to move people in before the building is completely done?	Legal HR Facilities Building contractor	Meeting	5

4.4 Prepare for Elicitation

Elicitation preparation may be formal or informal. Elicitation preparation is the planning performed to conduct an effective elicitation session. The business analyst may create informal preparation notes to organize and to help facilitate the session. Preparation notes can be used to measure the progress achieved in a session against what was planned to be achieved and can be used to adjust expectations for future sessions.

4.4.1 Determine the Objectives

To ensure elicitation activities are effectively performed, the business analyst should set an objective for each session to achieve. The objective is the reason why the elicitation activity is being undertaken. Each session should provide some value and benefit to justify the time it takes to obtain the needed information.

4.4.2 Determine the Participants

At the completion of stakeholder analysis, the business analyst should have divided the long list of stakeholders identified into groups or classes. The results of the stakeholder analysis can be used when selecting the participants to invite to an elicitation session. If additional stakeholders were uncovered after the stakeholder analysis was completed, the business analyst should update the stakeholder register to ensure there are no stakeholder groups overlooked and factor in elicitation time for these new groups.

The business analyst schedules the appropriate amount of time for each stakeholder group; it may be appropriate to schedule less time with executives than with end users.

4.4.3 Determine the Questions for the Session

When the objective of the elicitation activity suggests using interviews, focus groups, facilitated workshops, or other techniques used to elicit information directly from stakeholders, the business analyst may want to prepare some questions prior to conducting the elicitation in order to ensure the session objectives are achieved. It is often effective to identify more questions than needed in the event that questions are answered more quickly than expected.

The business analyst should know the rationale for each question and be able to explain why a question is important or why it is being asked. Questions that do not move the session forward to achieving the desired objectives or those questions that obtain data that do not have a specific purpose should be avoided.

Figure 4-1 is an example of preparation notes for an elicitation session to discuss an office move.

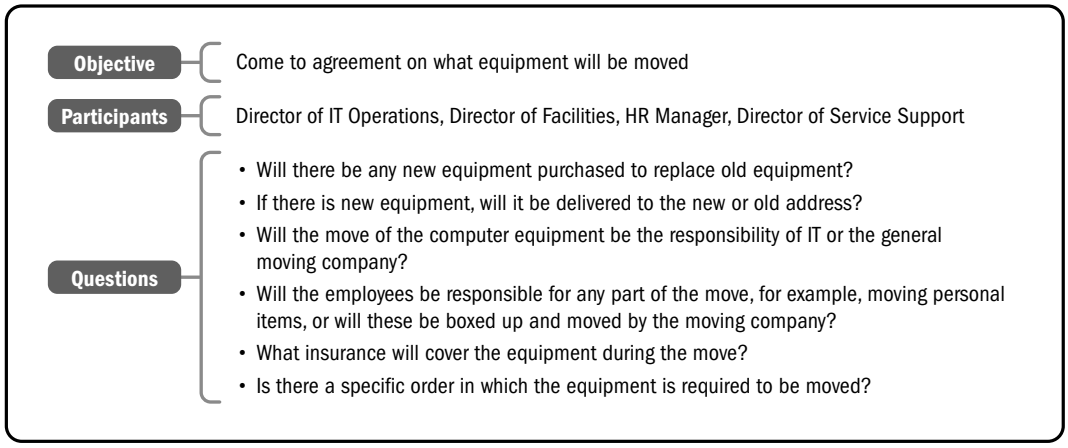


Figure 4-1. Example of Preparation Notes for an Elicitation Session

The questions in Figure 4-1 are in no particular order. The business analyst may organize the questions with easy, nonthreatening questions at the beginning and end, with the more challenging, complex, or contentious questions in the middle of the session. Doing so helps the group make progress in the session early before needing to address the challenging questions, as well as ensuring the session ends on a positive note.

Elicitation preparation may be formal or informal depending on the preference of the business analyst. Preparation notes are solely for the benefit of the business analyst to ensure elicitation is facilitated effectively.

4.5 Conduct Elicitation Activities

There are four stages during the actual elicitation activity in which information is gathered:

- **Introduction.** The introduction sets the stage, sets the pace, and establishes the overall purpose for the elicitation session.

- **Body.** The body is where the questions are asked and the answers are given.
- **Close.** The close provides a graceful termination to the particular session.
- **Follow-Up.** The follow-up is where the information is consolidated and confirmed with the participants.

The information provided in Sections 4.5.1 through 4.5.4 is applicable whether the elicitation session is conducted in the form of an interview, workshop, or focus group. Section 4.5.5 provides a list of techniques that may be used to conduct elicitation.

4.5.1 Introduction

During the introduction, the business analyst establishes the frame for the session and sets the tone and rapport with the participants. The frame is created at the beginning of each session by stating the problem and by providing an overview of the session objectives. The frame is a cognitive technique that causes the participants in a session to subconsciously focus on the subject at hand. The structure keeps the participants on track and thinking about the objectives. Stating the benefits that will be achieved when the objectives are met helps to increase stakeholder participation and provides relevance to the information that participants are sharing.

The business analyst should consider use of a "parking lot" as a tool to keep participants on track. Parking lots are used to minimize sidetracking, derailing, or hijacking of the meeting by participants. The parking lot, created on easel pads, white boards, or electronic equivalent, is a place to store topics that have been raised by the participants which do not relate to the session objectives. The business analyst should explain the session rules including the use of the parking lot prior to the start of the elicitation.

4.5.2 Body

The body of the elicitation session is where the business analyst's soft skills come into play: active listening, empathy, body language, selection of questions to ask, sequencing of questions, influencing skills, etc. During the body, the business analyst elicits the primary information and achieves the objectives of the elicitation session.

Moving from the introduction to the body ideally should be a seamless transition. Handled well, the participants should not notice the shift to the more challenging questions and should continue to answer questions. The body is the portion of the session where momentum is built and the team is highly performing and engaged in providing a large amount of information.

4.5.2.1 Types of Questions

There are a number of different types of questions that can be asked. The business analyst selects the appropriate types of questions to meet the objectives and to mix up the question types in order to generate the most information.

Not all questions are planned. Much of the conversation during elicitation, while directed and investigative in nature, is spontaneous. Through practice and experience, the business analyst develops the ability to probe and

further dig into the details and adjust the direction of questioning based on the responses received. The types of questions that can be asked are as follows:

- **Open-ended question.** A question that allows the respondents to answer in any way they desire.
- **Closed-ended question.** A question that calls for a response from a limited list of answer choices. Types of closed-ended questions are forced choice, limited choice, and confirmation.
- **Contextual question.** A question that requires an answer regarding the subject at hand; namely, the problem domain or the proposed solutions.
- **Context-free question.** A question that may be asked in any situation. Context-free questions are also used as lead-ins to obtain information to define the solution.

4.5.2.2 How to Ask the "Right" Questions

The right question is not known until after the question is answered and analyzed. There is no one right question that will provide the exact information that will generate the correct solution. In many cases, information that leads to the perfect solution arrives in pieces from many questions and many participants' responses.

The more questions asked, the greater the chance to ask the right question. One way to know whether the right questions were asked is to analyze the results of the elicitation and assess the outcomes to see if the objectives of the session were achieved.

4.5.2.3 Listening

The business analyst needs to employ active listening to capture all of the information that comes with each answer. Active listening is the act of listening completely with all senses. Active listening entails paraphrasing or reciting back what is heard to ensure an accurate understanding of what has been communicated. It involves suspending all judgment about what is being heard so that information flows freely. When discrepancies in information are identified, the business analyst should not call attention to the discrepancy, but instead continue to draw out other clarifying information through questioning. One goal of active listening is to clear up discrepancies without raising the possibility of conflict.

During elicitation, some stakeholders will not provide information that they believe the business analyst may already know. In these scenarios, the stakeholder may provide only high-level answers to the questions raised, assuming that the business analyst understands the lower-level details. In order to gain access to the specifics, the business analyst needs to communicate to the stakeholders that the information being provided is unknown and needed. This will ensure the flow of information is provided at a sufficient level of detail.

4.5.3 Close

The close occurs at the end of an elicitation session. The purpose of the close is to wrap up the activities and focus on next steps. When the elicitation conducted was in the form of a workshop, the business analyst may

consider a rundown of the assignment of action items so participants are aware of their responsibilities at the conclusion of the session. When the elicitation session was conducted in the form of an interview, the business analyst provides direction to the interviewee regarding next steps. The close is always used to thank participants for their contribution and time.

Three questions a business analyst may consider asking at the end of each elicitation session are as follows:

- Do you have any additional questions?
- Is there anything we missed or anything that we should have talked about but didn't?
- Is there anyone else who might have information that would contribute to the elicitation objectives?

Upon the conclusion of the elicitation session, the business analyst has a lot of information to process. As a result of analyzing this information, the business analyst may find that new questions emerge, ambiguities and contradictions surface, and clarity that was originally present turns to vagueness. This is not an uncommon outcome considering the large amount of information that is relayed during an elicitation session. The questions that arise during analysis become the materials to help structure the objectives for the follow-up sessions.

4.5.4 Follow-Up

In a follow-up session, after taking time to analyze and consolidate the information, the business analyst shares any revised notes and takes the opportunity to obtain confirmation from participants on the information obtained during the prior elicitation session. It is preferable to hold the follow-up after stakeholders have read the consolidated notes; however, business analysts will often find that participants do not have time to do so. The business analyst should use collaboration techniques to lead the participants through discussions to validate the consolidated information for accuracy and completeness before moving ahead.

A one or two paragraph summarization of the information elicited provides the following benefits:

- Provides an opportunity to fully analyze all the information received and to eliminate extraneous material;
- Allows time to verify and clarify the notes taken during the session, especially acronyms and abbreviations used in an effort to capture all of the information being provided;
- Uncovers any questions that should have been asked during the meeting;
- Reinforces to participants that their information is valuable enough for someone to spend the time to digest, analyze, and summarize it;
- Gives each participant a chance to respond to the summarization with additional information; and
- Provides participants with an opportunity to clarify or correct anything said previously.

  Note: An alternative approach business analysts can employ is to produce all notes on a whiteboard during the elicitation session. Attendees can clarify and correct information on the spot to avoid a read-review process. The business analyst determines the appropriate process based on the size of the project and the complexity of the information being elicited.

4.5.5 Elicitation Techniques

Some common techniques used for elicitation are as follows:

4.5.5.1 Brainstorming

Brainstorming is a technique used to prompt innovation and creativity by asking groups to consider novel or different solutions. Output generated from the group is often greater than the output from the same group when solutions are recorded individually.

For additional information on how to conduct brainstorming, refer to Section 3.3.1.1.

4.5.5.2 Document Analysis

Document analysis is an elicitation technique used to analyze existing documentation and identify information relevant to the requirements. Business analysts can start their analysis work with this technique to gain some understanding of the environment and situation prior to engaging directly with stakeholders.

Document analysis provides the following benefits:

- Information received from individuals is subjective, whereas documented information tends to be more objective. While it is always good to have the subjective viewpoints of a number of different individuals on the same topic, it is best to have the objective and factual information first to use as a baseline to understand the subjectivity variations.
- Documents may contain information that no one individual has. This is found in older descriptions of a system or business process, source material for regulations, and other mandatory procedures.
- Individuals may not have a totally accurate view of the information, whereas written documentation has been researched and is more accurate.
- Written documentation provides more background and explanations than an individual explaining the same material, because the individual may assume that the business analyst already knows much of the information or that common knowledge has already been documented elsewhere.
- Up-to-date documentation can be a good source of information regarding the structure and capabilities of any product, ranging from physical structures (e.g., buildings) to devices with embedded systems (e.g., cellphones, pacemakers, or thermostats) to software (e.g., a claims payment system). A suite of up-to-date regression tests may serve the same purpose for obtaining accurate information about software and embedded system functionality.

The downside of the document analysis method is that documentation may not be available or the existing documentation may be out of date, thereby providing erroneous information. Even when documentation is maintained and considered current, there is a risk that previous system constraints or limitations will be documented as current business practices. It may be difficult for the business analyst to decipher these limitations without having a current conversation with a business stakeholder.

4.5.5.3 Facilitated Workshops

- **Overview.** Facilitated workshops, also known as requirements workshops, are focused sessions that bring key cross-functional stakeholders together to define product requirements. Workshops are considered a primary technique for quickly defining cross-functional requirements and reconciling stakeholder differences. Due to their interactive group nature, well-facilitated sessions can build trust, foster relationships, and improve communication among the participants, which can lead to increased stakeholder consensus. In addition:
  - There is synergy when ideas from various people help to stimulate new thoughts from others.
  - Disagreements among business units or individual stakeholders are resolved as they come up during elicitation, saving time later.
  - Issues are discovered and resolved more quickly than in individual sessions.
  - Obtaining agreement on issues is easier when the group is assembled together.
  - Engagement is higher when stakeholders are urged to participate.
  - There is a perception that no one stakeholder will have a higher influence on the solution, because everyone is in the meeting together.

  The facilitated workshop may also include the solution team or its lead. The benefits of a cross-functional facilitated workshop are as follows:

  - There is a team building aspect that unites the group.
  - There is a better chance for a binding agreement between the solution team and the product stakeholders.
  - The solution team is more committed when they are able to meet directly with the stakeholders for whom they are building the solution.
  - The solution team or its lead learns the context of the problem, solution, and decisions, which provide a more informed basis for developing a solution.
  - The requirements resulting from a combined meeting are more likely to be implemented, because the work was developed collaboratively.
- **Workshop preparation.** Workshops are expensive to run when considering the number of individuals involved and the time commitments required for each attendee to participate. However, when run efficiently, workshops can produce a lot of results in a short time period. In order to ensure the workshop gets off to a strong start, the business analyst should prepare as follows:
  - Collaborate with the project team and stakeholders to agree on the agenda topics ahead of time.
  - Produce an agenda and distribute it with the workshop invitation. If it cannot be sent with the invite, provide at least two business days' notice.
  - When the workshop will involve decision-making activities, ensure the team has already determined their decision-making process prior to the start of the workshop. This information should have been

agreed upon during business analysis planning; however, in any event, it needs to be agreed upon before the start of the workshop.

- When any form of technology is being used to run the meeting, the meeting organizer needs to set up and start up all technical components prior to the start of the meeting. This may require that the organizer book the conference room 30 minutes to an hour prior to the start of the session so all technology is up and running prior to the start of the workshop. Examples of technology considerations are remote attendees dialing into a teleconference number, LCD projectors or web conferencing being used to display a presentation or notes, software used to obtain documents that will be discussed/reviewed, etc.
- Sufficient workshop materials are planned for and obtained prior to the start of the session. Stopping a workshop to retrieve materials loses the momentum.

- **Invite participants by roles.** When conducting a workshop or other form of elicitation within a group setting, invite attendees by role to avoid two persistent problems that often occur in elicitation sessions:
  - Interlopers or those who are not invited or do not need to attend the meeting.
  - Lack of attendance by those who are needed for the meeting but do not show.

  Instead of a broadcast invitation to everyone having anything to do with the project, the business analyst should direct the invitations only to the roles required to achieve the workshop objectives. Inviting by role establishes expectations for participation and encourages attendance. Participants are more apt to prepare and provide the information needed from the perspective of their role, and are less likely to miss the meeting because they now have a specific responsibility to play in the meeting.
- **Tips for running a workshop.** Introduce participants or have the participants introduce themselves by role.
  - Keep cross discussions and personal topics to a minimum to generate the most effective information.
  - Start the meeting on time to send a message that everyone's time is important. To maintain momentum and focus, do not stop the meeting for latecomers but instead brief them during the break.
  - Consider assigning separate roles, including a facilitator, a scribe, and a workshop owner to ensure a smoother session. It is a good practice to write directly on a whiteboard or some form of electronic medium so that the attendees can review the notes during the workshop. This reduces the need to conduct multiple reviews of the workshop results.

  It is challenging to lead a session while controlling the meeting and participants. Facilitators help the business analyst and the meeting attendees to achieve the meeting objectives. The scribes help to ensure that the results are documented and the workshop owner makes the business decisions. Separating the roles often results in collecting more information as the practitioner is less distracted having to deal with multiple responsibilities.

  Collaboration Point—It is not always practical for multiple business analysts to be present in a single facilitated workshop. This is an opportunity for the project manager and the business analyst to team

up. The project manager may support the effort by fulfilling one of the roles, such as the scribe, thereby allowing the business analyst to focus on the questions and elicitation of information.

- **Closing tips.** Make it clear to the participants when the elicitation session will end. If there is a question and answer portion of the workshop, ensure that the participants know how many more questions can be raised and responded to. This allows for a nice transition to wrap up the meeting and does not leave attendees concerned that their questions will go unanswered.
- **Follow-Up Tips.** Schedule time to thoroughly analyze the information received as soon after the workshop as possible. Information and context may be lost when the meeting notes are not completed soon after the working session. Try to provide the summarized results back to the participants in a time frame that meets the participant's expectations. The participants will benefit from reviewing the session results soon after the meeting closes to ensure their recollection of the discussions is not forgotten.

  Note: In today's project environment, there is more emphasis on collaboration and requirements discovery through team approaches. Unlike traditional meetings of the past, information does not simply flow from participant to facilitator, but can also flow among participants. Stakeholders may require meetings to discover from each other before they are able and ready to provide requirements information to the business analyst.

4.5.5.4 Focus Groups

A focus group is an elicitation technique that brings together prequalified stakeholders and subject matter experts to learn about their expectations and attitudes about a proposed product, service, or result.

Focus groups are used to gain feedback on completed work or prototypes. Generally, group members are prequalified or prescreened to ensure they meet the desired or targeted representation. A group size of 8 to 12 is optimal. The person chosen to facilitate the session should have experience moderating sessions of this type. The facilitator is often provided with an outline or a list of objectives to achieve in the session. Sessions are facilitated in a manner that allows for healthy team dynamics, a free flow of ideas, and a sufficient level of feedback to meet the session objectives.

Focus groups work well to allow participants to share ideas and build off of the feedback that is being shared among the group. The business analyst should watch reactions, facial expressions, and body language in addition to taking in the information being provided by the group. Due to their format and structure, focus groups are not a suitable method for eliciting information about a problem domain. One drawback of a focus group is that participants may be influenced by group pressure to agree with the stronger-willed participants. The facilitator plays an important role to engage the entire group and to ensure no one participant is demonstrating signs of being influenced by group pressure.

4.5.5.5 Interviews

An interview is a formal or informal approach to elicit information from stakeholders. It is performed by asking prepared and/or spontaneous questions and documenting the responses. Interviews are often conducted on an

individual basis between an interviewer and an interviewee, but may involve multiple interviewers and/or multiple interviewees. Interviewing experienced project participants, stakeholders, and subject matter experts helps to identify and define the features and functions for the desired solution.

Interviews work well when:

- An individual has been identified as being able to provide a variety of information on a variety of topics.
- Confidential or sensitive information is needed that should not be discussed in an open forum.
- Information needs to be acquired from an upper-level manager.
- The business analyst needs to probe deeply into the issues surrounding the problem or problem domain and needs unfettered access and time from a specific individual who may have that information.

There are two basic types of interviews:

- **Structured interviews.** Begin with a list of prepared questions with the goal of asking and obtaining answers to every question on the list or within the allotted time.
- **Unstructured interviews.** Begin with a list of prepared questions, but the only question that is definitely asked is the first. After that, the subsequent questions are based on the answers to the previous questions. The interview takes on a life of its own and requires skill to keep the information focused on reaching the objective.

Interviews may be conducted synchronously or asynchronously:

- **Synchronous Interviews.** These interviews are performed live or in real time. These can be conducted in a number of ways, such as face-to-face where the business analyst and the interviewee are in the same room, or they can be conducted over the telephone, with video conferencing, internet collaboration tools, etc. The key is that the interview is being conducted with the interviewee and interviewer at the same time.
- **Asynchronous interviews.** These interviews are not conducted in real time; the business analyst or interviewer is not involved in the interview at the same time as the interviewee. Asynchronous interviews can be conducted through email or recorded, for example, where the interview questions are scripted and the interviewee records their responses to the questions through video.

There are a number of advantages to conducting an interview in person, for example:

- Having the undivided attention of the interviewee,
- Ability to see the body language and facial expressions accompanying the answers, perhaps more clearly than with video conferencing or collaboration tools, and
- Providing a more comfortable setting for the interviewee, especially if they are discussing sensitive information, for example, flaws with the current system.

With global organizations and international project teams, face-to-face elicitation activities are often replaced by teleconferences and videoconferences. There are a number of benefits associated with virtual meetings, including time and cost savings as well as the ability to bring people together from all over the world, which would not be possible otherwise.

Some of the disadvantages associated with conducting virtual interviews are:

- Interviewees multitask during the session, resulting in lost information;
- Participants call into the session from various locations, such as airports, parking lots, other client sites, etc., with the accompanying distractions;
- Lack of experience on the part of the interviewer and the interviewee participating in virtual meetings; and
- Equipment failure or poor performance of collaboration tools.

All of these obstacles can be overcome with proper training and with the establishment of ground rules for conducting virtual meetings. The use of virtual meetings to elicit requirements is likely to increase in the future due to the substantial cost and time savings that this approach provides.

4.5.5.6 Observation

Observation is a technique that provides a direct way of viewing people in their environment to see how they perform their jobs or tasks and carry out processes. It is particularly helpful for detailed processes when people who use the product have difficulty or are reluctant to articulate their requirements. Observation is also called job shadowing. It is usually performed externally by an observer who views the business expert performing his or her job. It can also be performed by a participant observer who performs a process or procedure to experience how it is done so as to uncover hidden requirements. The objective of the technique is to elicit requirements by observing stakeholders in their work environment.

Observation often results in the transfer of a greater amount of unbiased, objective, real information about the problem domain than with other forms of elicitation. When asked in a meeting to describe how to go about performing their work, it is probable that a stakeholder may miss steps or undercommunicate.

Cognitive experiments have shown that the more experienced a worker is, the more difficult it is for them to understand why specific decisions are made when performing a task. Through observation, the tasks and decisions being made are seen firsthand, and questions may be asked directly about why a particular decision is being made.

In business analysis, the business analyst fulfills the role of observer or participant observer. Observation in business analysis is not used to evaluate the performance of the worker, that is, to collect performance related information for input into a performance evaluation process. When the worker is suspicious of the observer, the value of observation may be diminished.

Additional benefits of observation are:

- Provides more insight into tasks and activities that are difficult to describe,
- Provides an opportunity to request a demonstration of complicated tasks or specific interactions with a system or product and to obtain an explanation of the process being performed,
- Provides information and visualization together, and
- Provides context around the activity.

There are four types of observation; each is used in a different situation:

- **Passive observation.** The business analyst observes the worker at work without interrupting, asking questions, or seeking clarification. The observer records the observations, often in the form of a process flow with timings recorded on the diagram. At a later date, the business analyst can ask the worker about the activities observed in order to clarify and validate notes. An advantage of passive observation is the minimal interruption to the workflow. As a result, an organization may not allow any other form of observation to be conducted, especially by an outsider. A disadvantage is that the worker may not trust the observer and may perform the work in a nonroutine fashion.
- **Active observation.** During active observation, the observer interrupts the process or activity, asks questions about what the worker is doing, seeks clarification, and asks for opinions, etc. The advantage of active observation is in the immediacy of information elicited and the increased amount of information collected. Active observation does, however, interrupt the flow of work which reduces productivity and possibly changes behaviors during the observation.
- **Participatory observation.** During participatory observation, the observer takes part in performing the activities being observed. It allows the observer to generate questions that would never have been thought of otherwise. In addition, the observer has an opportunity to experience what workers are going through when they perform these activities. The observer may discover functions, features, and improvements that would never come up during a facilitated workshop or interview.
- **Simulation.** The observer simulates the activities, operations, or processes of the work using a tool that recreates the work of a process worker in a simulation. The organization may have training facilities where the observer can interact with test versions of a system or product. With simulation, the business analyst follows up with the process worker through an interview to elicit further details.

The main drawback to observation is that people act differently when they are being observed. In addition, some managers may not support interruptions or may not allow firsthand observations. When observations are used, the information obtained should be reviewed and validated by the person who performed the demonstration to ensure accuracy and avoid any bias or assumptions that the business analyst may have introduced in the observation notes.

4.5.5.7 Prototyping

Prototyping is a method of obtaining early feedback on requirements by providing a working model of the expected product before building it. Since prototypes are tangible, stakeholders are able to experiment with a model of the final product rather than discussing abstract representations of the requirements. Prototypes support the concept of progressive elaboration in iterative cycles of mockup creation, user experimentation, feedback generation, and prototype revision. A prototype can be a mockup of the real result as in an architectural model, or it can be an early version of the product itself.

Elicitation and thorough investigation may not uncover all of the attributes or aspects of a complex solution. Allowing the users and customers to see the product or system as it is being built provides an opportunity for the business to identify issues, clarify requirements, and provide additional information that may have been omitted originally.

The key element to prototyping is the iterative process of creating the prototype, reviewing it with the pertinent stakeholders, making adjustments and modifications to the prototype, and reviewing it again. This process continues until all parties agree that the prototype has provided the needed information to the solution team.

There are two types of prototypes: low-fidelity prototypes and high-fidelity prototypes.

- **Low-fidelity prototype.** Low-fidelity prototypes are completed with a pen and paper, marker and whiteboard, or modeling tool on the computer. Examples of low-fidelity prototypes include:
  - Wireframes,
  - Mockups of interface screens or reports,
  - Architectural renderings of a building,
  - Floor plans,
  - Sketches of a new product, and
  - Any design that is evolving.

  A typical use for a low-fidelity prototype is to mock up user interface screens and share them with the intended users to provide a visual representation of what the product/solution will look like and how it will function.
- **High-fidelity prototyping.** High-fidelity prototypes create a representation of the final finished product and are usable by the stakeholders who are reviewing them. Typically, the prototype has limited data, is restricted to a single computer device, and has partial functionality. High-fidelity prototyping is performed in an iterative fashion. Reviewers can manipulate the screen, enter some data, and move from screen to screen to experience firsthand how the screen will work.

  There are two types of high-fidelity prototypes: throwaway and evolutionary.
  - *Throwaway prototypes* are discarded once the interface has been confirmed. This is similar to the product prototypes developed by manufacturing companies. The prototype is used to help define the tools and process for manufacturing the product but the prototype itself is not sold.
  - *Evolutionary prototypes* are the actual finished product in process. The first prototype that is reviewed is the earliest workable version of the final product. With each successive prototyping session, more functionality is added or the existing functionality is modified to achieve a higher level of quality.

  Note: With agile project teams, evolutionary development is how the product is built. The work is not considered to be a prototype but is an actual slice of the product itself.

Two examples of prototyping techniques are:

- **Storyboarding.** Storyboarding is a prototyping technique showing sequence or navigation through a series of images or illustrations. In software development, storyboards use mockups to show navigation paths through webpages, screens, or other user interfaces. Storyboards are graphical representations

that represent the sequence of events. Storyboards are typically static and thrown away. Prototypes focus on what the product will look and feel like, while storyboards focus on the experience.

- **Wireframes.** A wireframe is a diagram representing a static blueprint or schematic of a user interface and is used to identify basic functionality. A wireframe separates the look and feel of a site from its function. It presents a stripped-down, simplified version of the page, devoid of distractions. The purpose of the wireframe is to illustrate the flow of specific logical and business functions by identifying all entry and exit points or actions the users will experience. A typical wireframe contains:
  - Key page elements such as header, footer, navigation, content objects, branding elements, and their respective locations;
  - Groupings of elements such as side bars, navigation bars, and content areas;
  - Labeling, page title, and navigation links; and
  - Placeholders, content text, and images.

  Wireframes can be developed as a basic blueprint with low fidelity and may serve as an inexpensive approach to gleaning design preferences from stakeholders. Wireframes drive communication and help to support evolutionary discovery of requirements.

4.5.5.8 Questionnaires and Surveys

Questionnaires and surveys are written sets of questions designed to quickly accumulate information from a large number of respondents. Respondents represent a diverse population and are often dispersed over a wide geographical area. This method is beneficial for collecting a large amount of information from a large group over a short period of time at a relatively small expense. However, questionnaires and surveys also have these disadvantages:

- There is no opportunity for clarification, which could render the answers meaningless. In a face-to-face meeting, questions can be clarified when the answer is not what is expected.
- The formulation of the questions are often closed-ended, which could also render the answers meaningless.
- The number of responses the business analyst receives may not be significant enough to serve as a representative sample. A return rate of approximately 4 to 7% is not uncommon for a survey conducted within an organization. This means that without careful consideration of the survey sample, the information may be significantly prejudicial and draw the wrong conclusions.

For example, respondents who have an extreme reaction to a subject will tend to answer the survey to promote their ideas or concerns. When there is a small return on a survey that was distributed to a wide group of people, the results may only reflect those with an avid interest in a certain aspect of the survey, thereby skewing the results unfavorably.

Some ways of limiting the risks associated with surveys are as follows:

- Determine the number of respondents that will be required. Use a sample size calculator to determine the sample size needed for a statistically significant response.

- Analyze for skewed information when the surveys are returned. This will require that the survey includes some type of demographic or segmenting question to enable this analysis.
- Share information regarding why the survey is important to the organization and project.
- Send out reminders during the survey window to encourage and promote participation.
- Ask an upper management representative to champion the effort and emphasize its importance.

Make the survey results available to those who participate in the process as a follow-up and thank them for taking the time to complete the survey.

4.6 Document Outputs from Elicitation Activities

It is important to document the results of elicitation activities, either formally or informally. Documentation can range in formality from snapshots of whiteboards to fielded information recorded in requirements management tools. The primary documented result is a set of elicitation notes comprised of a wealth of information for performing other business analysis tasks. The results may come in the form of sketches, diagrams, models, flipcharts, sticky notes, or index cards, to name a few. When the elicitation results are analyzed, the results are documented into the deliverables and forms geared to the audiences who will use them.

4.7 Complete Elicitation

The elicitation process is an iterative process of alternating the steps of eliciting information and analyzing the information obtained. It can be considered a progressive elaboration of information. When information is analyzed, the quantity sometimes decreases, because extraneous information is removed. However, when the results are vague and open to interpretation, additional questions need to be asked and more elicitation sessions are then conducted.

A typical business analysis quandary is determining when the elicitation stops and the analysis starts and for how long the work continues. Analysis generates additional questions to clarify ambiguities, solidify vagueness, and add in more information about a particular topic, etc. This results in another round of elicitation activity, which increases the quantity of information, followed by another round of analysis decreasing the quantity of information. This continues until the analysis produces no further questions and the information is reduced down to a depiction of the solution to the business problem or when the risk of problems emerging from the lack of complete information is considered to be acceptable; and that is when elicitation ends.

> **Note:** Projects using an adaptive life cycle need to plan sufficient time for elicitation and analysis, but analysis is not usually estimated separately. Instead, analysis is taken into consideration and understood to be part of an estimate for the delivery of features or functionality within an iteration. Analysis occurs throughout the project as part of defining the initial backlog, grooming the backlog as the project moves forward, and analyzing details for each iteration.

The following may indicate when sufficient information has been elicited:

- The stakeholder or problem owner approves the results.
- The model on which the information is based can be completed.
- A dry run or successful prototype is completed.
- The objective has been reached.
- The solution(s) has been identified.
- Stakeholders begin repeating themselves and providing redundant information.
- It takes longer to get answers out of the same stakeholders. Stakeholders are trying to come up with new information that is different from what they previously gave, because elicitation is still in process.
- All information pertaining to high-priority requirements has been confirmed by at least two independent sources.

  Note: For a project following an adaptive project life cycle, the project team pushes toward replacing up-front analysis in favor of eliciting details within the increment in which the features will be delivered.

4.8 Elicitation Issues and Challenges

There are a number of difficulties associated with elicitation, for example:

- Conflicting viewpoints and needs among different types of users,
- Conflicting information and resulting requirements from different business units,
- Unstated or assumed information on the part of the stakeholders,
- Stakeholders who are resistant to change and may fail to cooperate and possibly sabotage the work,
- Inability to schedule time for interviewing or elicitation sessions because stakeholders cannot take time away from their work,
- Inability of stakeholders to express what they do or what they would like to do, and
- Inability of stakeholders to refrain from focusing on a solution.

The following lists some elicitation challenges and a few suggestions for meeting those challenges:

- **The business analyst cannot gain access to the right stakeholders.** A common issue business analysts face during the elicitation process is the inability to directly interact with the actual users of the product or system. As a result, user interfaces or processes may need to be developed without access to or input from those who are directly impacted by the proposed changes.

  The business analyst can address this issue by focusing on the information not the individual. Sometimes the desired information is available from multiple sources, for example, documentation, training materials,

operating procedures, etc. The business analyst may look to these other sources when gaining access to stakeholders proves to be difficult but needs to realize that doing so comes with great risk. The sponsor, project manager, and project team need to be fully aware when these situations arise, because moving ahead without the right stakeholders often leads to solutions that the business never accepts.

- **Stakeholders do not know what they want.** It can be frustrating trying to elicit requirements from stakeholders who do not know what the product is that they are asking for. There is a sense that the business stakeholders want to try various features until something works, but there is schedule pressure to produce sufficient requirements for the solution development team.

  To address this challenge, using techniques such as prototyping or storyboards may help the stakeholders visualize each of the possible solutions. The project team may consider using an adaptive project life cycle since it a preferred approach when there are changing customer needs or when stakeholders need to visualize the solution to further define their requirements.

- **Stakeholders are having difficulty defining their requirements.** Many business stakeholders come to elicitation activities with a solution in mind, and it may be difficult for them to describe the business problem they are trying to solve.

  To address this challenge, the business analyst should ask the business stakeholders for help in understanding the problem domain and focus their attention on the problem or opportunity they wish to address. After clarifying the situation, the discussion should be focused on the high-level requirements. When the business analyst helps to break down the high-level requirements and walks the stakeholders through the process, the stakeholders will be prevented from moving directly to the solution. When it becomes difficult to elicit needs and high-level requirements from the stakeholders, the business analyst needs to continue to ask clarifying questions to draw out the requirements.

- **Stakeholders are not providing sufficient detail to develop the solution.** Stakeholders may not have experience providing requirements or do not understand the business analysis process or business analyst role. Sometimes stakeholders are unaware of the level of detail that the business analyst wants to know or cannot figure out how to go about describing such details.

  To address this issue, the business analyst should try to elicit requirements through visual modeling techniques. Engaging stakeholders in modeling can open up dialogue that may not be possible through interview questions, surveys, or straight discussion. Ask stakeholders to help complete a workflow or to assist with breaking down a problem into a hierarchical model. This process focuses the stakeholder on completing the visual elements, resulting in discovery of details that might not be possible to obtain without the imagery.

4.9 Analyze Requirements

4.9.1 Plan for Analysis

4.9.1.1 Analysis Defined

One of the primary activities that business analysts perform is analysis, with much effort focused on requirements-related information. Analysis is the process of examining, breaking down, and synthesizing

information to further understand it, complete it, and improve it. Analysis entails looking closely at the parts of the information and how they relate to one another. Analysis involves progressively and iteratively working through information to lower levels of detail and often entails abstracting to higher levels of detail. Analysis is used to provide structure to the requirements and related information.

4.9.1.2 Thinking Ahead about Analysis

Planning for analysis involves thinking about what activities and techniques are likely to be useful and when they should be used. Not all techniques need to be decided before analysis begins, but by thinking ahead, it is more likely that business analysts will be prepared to use a variety of techniques.

Part of planning for analysis includes determining which types of models would be most beneficial given what is known about the project. There are a number of analysis techniques to choose from, and each technique has strengths and circumstances to which it is best used. A business analyst will likely not be proficient in every technique; however, it is valuable to learn as many techniques as possible and develop the skills and experience to know when a particular technique is best leveraged. Consider which analysis tools could be applied, such as modeling tools, and which templates could be used. Decide in advance which modeling language to use, if any. Determine what existing models in the organization could be used as a starting point for the current project. Plan for analysis based on known information but also be able and ready to adjust plans to the unexpected discoveries that occur throughout the business analysis process.

4.9.1.3 What to Analyze

Business analysts work with different types of information. Choose the correct information to analyze and separate out those things that interfere with a proper analysis. Use visual models to help establish a boundary on exactly what needs to be analyzed and to help facilitate discussions with stakeholders and subject matter experts when determining the key pieces of information. Most often, business analysts conduct analysis on the outputs of elicitation activities. In addition, analysis frequently provokes relevant and important questions about the situation, requiring more elicitation. Regardless of the business analysis approach used, elicitation and analysis are usually iterative.

4.10 Model and Refine Requirements

4.10.1 Description of Models

In this practice guide, the model refers to a visual representation of information, both abstract and specific, that operates under a set of guidelines in order to efficiently arrange and convey a lot of information in a concise manner. In its simplest form, a business analysis model is a structured representation of information. Models are diagrams, tables, or structured text. Models are created with a variety of tools, ranging from formal modeling tools to whiteboards to artistic software. There are many business analysis models and each serves one or more purposes. Examples of information related to business analysis that can be modeled include business objectives, requirements, business rules, and design.

4.10.2 Purpose of Models

Business analysis models are helpful to find gaps in information and to identify extraneous information. Models provide context to better understand and more clearly convey information. Requirements are modeled and refined to achieve further clarity, correctness, and to elicit additional information to define the details necessary for the product to be built.

When the correct models are applied, analysis becomes simple relative to analyzing the information in pure text form, because the models help visualize and summarize complex information. When the correct models are applied, analysis becomes much easier.

4.10.3 Categories of Models

Analysis models are organized into specific categories, defined mostly by the primary subject matter represented. One categorization of models is shown in Table 4-2, along with examples of each model.

Table 4-2. Models Organized by Category

Category	Definition	Example Models
Scope models	Models that structure and organize the features, functions, and boundaries of the business domain being analyzed	▪ Goal and business objectives model ▪ Ecosystem map ▪ Context diagram ▪ Feature model ▪ Organizational chart (described in Business Analysis Planning) ▪ Use case diagram ▪ Decomposition model (described in Business Analysis Planning) ▪ Fishbone diagram (described in Needs Assessment) ▪ Interrelationship diagram (described in Needs Assessment) ▪ SWOT diagram (described in Needs Assessment)
Process models	Models that describe business processes and ways in which stakeholders interact with those processes	▪ Process flow ▪ Use case ▪ User story
Rule models	Models of concepts and behaviors that define or constrain aspects of a business in order to enforce established business policies	▪ Business rules catalog ▪ Decision tree ▪ Decision table
Data models	Models that document the data used in a process or system and its life cycle	▪ Entity relationship diagram ▪ Data flow diagram ▪ Data dictionary ▪ State table ▪ State diagram
Interface models	Models that assist in understanding specific systems and their relationships within a solution	▪ Report table ▪ System interface table ▪ User interface flow ▪ Wireframes ▪ Display-action-response

4.10.4 Selection of Models

Choosing the correct model can be difficult because often there are multiple valid choices. It is unlikely that all of the models will be used on one project, but for most projects, more than one type of model will be used. In some cases, there are many models that could be applied, but time constraints may require the business analyst

to choose only a few of the models. In these scenarios, models need to be prioritized according to applicability. The business analyst should consider the following when choosing which models to use:

- **Methodology.** The choice of models or formality and depth of models can be methodology independent. However, certain models are more suited to one methodology than another.
- **Characteristics of the project.** Project characteristics, such as, business process, automation, custom development, commercial-off-the-shelf, cloud or software as a service, data migration, workflow, mobile, hardware, software, number of users, analytics, and reporting are considerations when selecting the correct models for the project.
- **Timing within the project life cycle.** Some models are better used early in a project when defining the project's value and scope, or when identifying stakeholders. Other models are more appropriate as the project progresses and the low levels of details are being described.
- **Categories of models.** Models from every category (see Table 4-2) should be considered on every project.
- **Level of abstraction.** Models represent different abstraction levels. Some are better suited for analyzing a whole solution, others for elements of a solution, and others for details within an element.

For example, an agile project will likely use user stories as opposed to use cases. A reporting or analytics project will likely use data models, including a data dictionary and report tables. A system migration project will probably have scope models such as an ecosystem map or context diagram, as well as data models like a data dictionary and a business rules catalog. Early in a project, business analysts usually create context diagrams, ecosystem maps, and high-level process flows. Later in a project, business analysts may create state models, decision models, and user interface models. A project that involves automating operational functions benefits from process models to elicit information about how work is currently conducted and how work will be performed after the automation is implemented.

It is helpful to use more than one model, because the models complement one another and enable analysis of the project from different perspectives. For example, a data dictionary describes attributes of the business data objects and those data objects are also reflected in an entity relationship diagram. The allowed transitions on a state diagram are reflected in one or more process models. Cross-checking models against each other will help find gaps, unnecessary information, and potentially missing requirements.

4.10.5 Use Models to Refine Requirements

Business analysts use models to determine what is important and valuable so that the right requirements are created. Models are used during elicitation sessions to refine requirements with stakeholders or subject matter experts. Through an iterative process, the details become sharper and sharper until there is a clear enough picture as to what is important and what is not. Each model explained in Sections 4.10.7 through 4.10.11 describes how the model helps to identify and refine requirements or how the model relates to requirements.

4.10.6 Modeling Languages

There are many modeling languages, and each has its strengths and weaknesses. Some common modeling languages used in business analysis are described in Table 4-3.

Table 4-3. Modeling Languages and Usage

Modeling Language	Overview of Usage
Business process modeling notation (BPMN)	Used to model complex business processes for the purpose of making changes to these processes.
Requirements modeling language (RML)	Used to visually model requirements for easy consumption by all stakeholders, particularly business stakeholders.
System modeling language (SysML)	Used to analyze complex systems and includes a subset of UML.
Unified modeling language (UML)	Primarily used to specify design models but can work well to specify requirements.
Various other modeling languages	Used when a specific modeling language isn't appropriate or not part of the organizational standards. For example, process models are frequently created using ISO-standard flowchart symbols. Data models often use Information Engineering "crow's foot" notation.

Whether a specific modeling standard is used during analysis or not is unimportant; what is important is to use consistent syntax each time a similar model is used so as not confuse stakeholders. For example, when creating process flows, use the same shapes to mean the same things. In addition, it is helpful to keep the models as simple as possible. It is difficult for stakeholders to read and understand models that contain overly complex syntax and information. Some organizations develop guidelines and standards to ensure consistency across the organization. It is helpful to add a key or legend to a model to ensure that everyone understands what the symbols represent.

Sections 4.10.7 through 4.10.11 describe a variety of models from each category. Each model description includes an overview of the model and its syntax, an example, common ways it is used, and how the model relates to requirements.

> *Example*—The examples for the models are based on a mock project called "recipe box" for a grocery chain. This is a mobile application project that allows users to select recipes, look up grocery stores that have the ingredients, and then map the location of the ingredients in the store. A new recipe is sent daily by email to participating customers. Ingredients for the recipe are on sale at each of the grocery stores. Customers can use a mobile device to run the Recipe Box application to display the current and past recipes. The application also shows the location of the items for the recipe at the store selected by the customer.

4.10.7 Scope Models

In general, scope models are used to structure and organize the goals, objectives, features, functions, and boundaries of the business domain being analyzed.

4.10.7.1 Goal Model and Business Objective Model

Goal models and business objective models are diagrams for organizing and reflecting goals, business problems, business objectives, success metrics, and high-level features. Chains of business problems and business objectives easily show where the project value comes from. Whether the value is identified as increasing revenue, decreasing cost, or avoiding penalties, goal models and business objective models visually represent the value that supports feature prioritization decisions and product scope management.

> ***Collaboration Point***—The project manager may be able to help complete portions of the goals and business objectives. For example, when the project manager has already developed a cost-benefit analysis or business case, some of the information needed in this model may already exist.

Example—Figure 4-2 shows one form of business objectives model for the Recipe Box project. In this example, the high-level features trace back to the overarching business problem that is being addressed—profit per visit is down. The measurable business objective is to increase profit per visit by $3 (a goal determined by the business group). To increase profit per visit, the grocery chain wants to increase sales of items that are more profitable. The Recipe Box product includes features that increase the consumer's desire to purchase such items and also make it easier for them to locate and purchase the items.

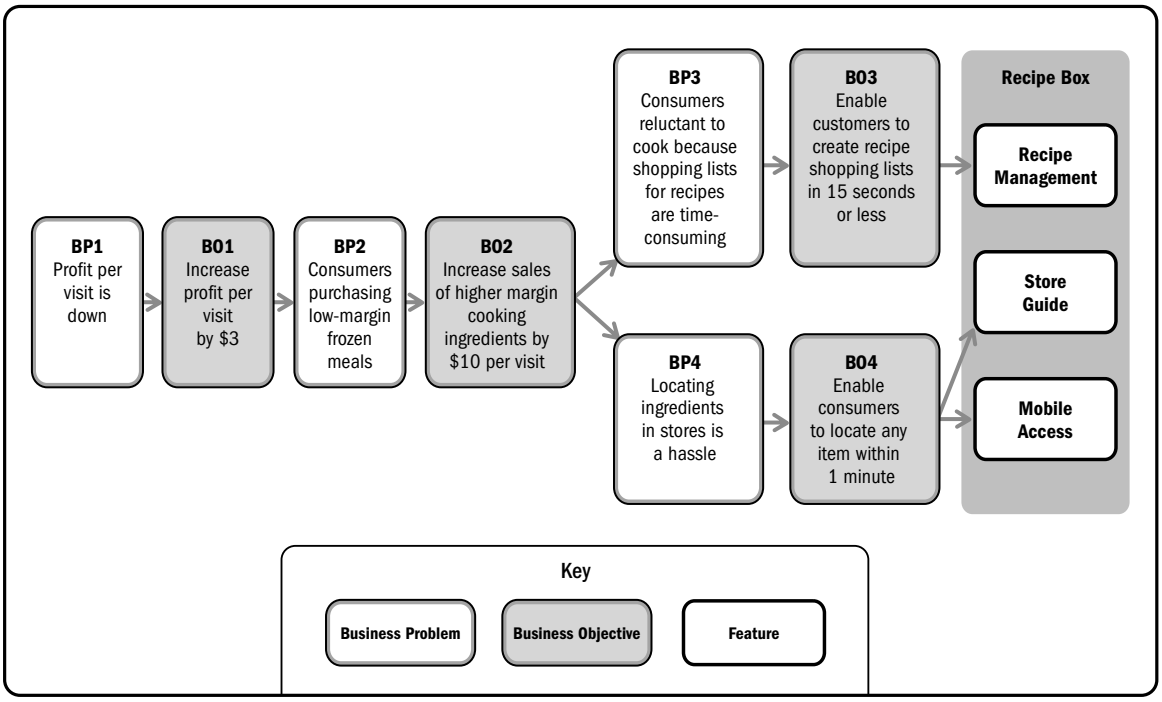


Figure 4-2. Example Business Objectives Model

- **Usage.** Although commonly constructed during the planning phase, a goal model or business objectives model can be created at any time during the project. It may be helpful to create the model as soon as possible so that teams and business owners can start assigning the numbers to the features they are attempting to develop for a particular product. It can be used to justify budgets as well as to reveal to executives exactly what they are receiving from a project. When business objectives are mapped to the requirements, scope control becomes much easier as the particular value of a specific requirement is better understood.
- **Relationship to Requirements.** These models provide a structure to specify business requirements. Each functional requirement produced for a project should be traceable to the identified business problems and objectives. Maintaining a focus on the top-level business problems and business objectives guides the delivery of valuable solutions and shapes the scope of the features. The value of requirements or features can be quantified based on how these requirements contribute to the business objectives in the model; this helps to identify the most important features to implement or identifies the minimally marketable features (MMFs).

4.10.7.2 Ecosystem Map

An ecosystem map is a diagram that shows all the relevant systems, the relationships between them, and optionally, any data objects passed between them. The systems are logical systems and therefore may not match an actual architecture diagram of physical systems. An ecosystem map is made up of boxes representing the systems and lines between the boxes that depict the relationships. When data is shown in the diagram, the labels on the lines identify the data objects and the arrows show the direction that the data flows. A system should be depicted in an ecosystem map when it is in scope for the project. It should also be shown when it is a system that passes or consumes data used by or manipulated by the systems in the project.

Example—Figure 4-3 is an ecosystem map for all of the systems within the grocery store solution and their interactions. This ecosystem map contains all of the systems that operate within a single grocery store, including external systems that transfer data. Although some of the systems do not directly interact with the Mobile App, the data generated or transferred can interact with the mobile application by means of an intermediary system.

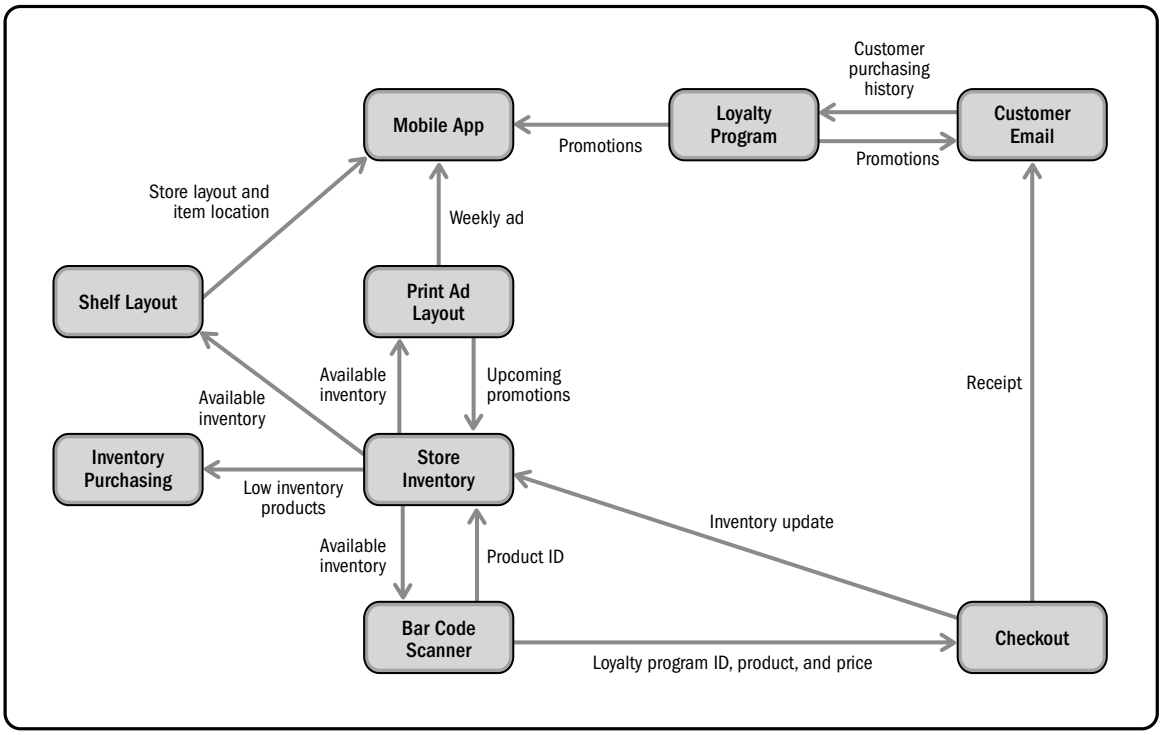


Figure 4-3. Example of Ecosystem Map

- **Usage.** An ecosystem map is used to understand all of the systems that may be affected by or that will impact the in-scope systems. It is a good model to represent systems that are in scope early in a project. In particular, the model is used to determine where there are possible interface requirements or data requirements.

  This model is slightly different than a context diagram, because ecosystem diagrams may include interfaces and systems that the solution under analysis does not interact with directly.

Example—When a mobile application is being developed and there is a shelf layout system that directly interacts with the mobile application, both systems should appear in an ecosystem map and a context diagram. However, there could be an inventory system that does not interface with the mobile application, but does interface with the shelf layout system. In this case, a business analyst may decide to show the inventory system in an ecosystem map so the audience understands the source for the inventory data that the shelf system is using. A context diagram would not show the inventory system, because it does not directly interface with the mobile application. Using the ecosystem map, a business analyst considers how the systems relate and sees interactions that may otherwise have been overlooked. Context diagrams are explained in Section 4.10.7.3.

- **Relationship to Requirements.** An ecosystem map is a high-level representation of system interfaces, but it does not contain specific requirements about these interfaces. System interface tables should be completed for each of the interfaces identified in an ecosystem map. Data models should be created to define data requirements for each of the data objects passed between the systems.

4.10.7.3 Context Diagram

A context diagram shows all of the direct system and human interfaces to systems within a solution. The diagram shows the in-scope system or systems and any inputs or outputs including the systems or actors providing or receiving them. A context diagram typically shows the system under development in the center as a circle, interfacing systems as boxes, human actors as people shapes or boxes, and lines connecting them to show the actual interfaces and the data passed between them. Context diagrams are sometimes referred to as Level 0 of a data flow diagram. Data flow diagrams are further discussed in section 4.10.10.2.

Example—The context diagram in Figure 4-4 displays the interactions between the Mobile App system being developed and external entities. It is similar to the ecosystem map in concept, because it illustrates data transfer or interactions between systems, but it includes more entities than just systems, such as the grocery shopper, and it only includes entities that directly interact with the system being developed. For example, the inventory system sends information to the shelf layout system. Both systems are included in the ecosystem map, but only the shelf layout system is in the context diagram as it has a direct interface.

- **Usage.** Context diagrams are particularly useful early in a project to specify the scope of the project, including any interfaces that have to be developed. It also shows all of the external touch points between the system under development and other systems or people. Context diagrams are also helpful in determining where there could be interface requirements or data requirements. Context diagrams have begun to be used by business analysts more broadly (i.e., to model business, user, and data contexts), because context diagrams are easy to build and understand. Context diagrams can also model the as-is and the to-be states in order to help with gap analysis.
- **Relationship to Requirements.** Context diagrams do not specify requirements but summarize the product scope and related information that are analyzed to identify requirements. Because context diagrams are used to specify all of the interfaces, these diagrams can help identify when interface requirements need to be elicited, for example, system and human interface requirements. This model often leads a business analyst to create system interface tables, user interface flows, display-action-response models, or other interface models that help to specify interface requirements.

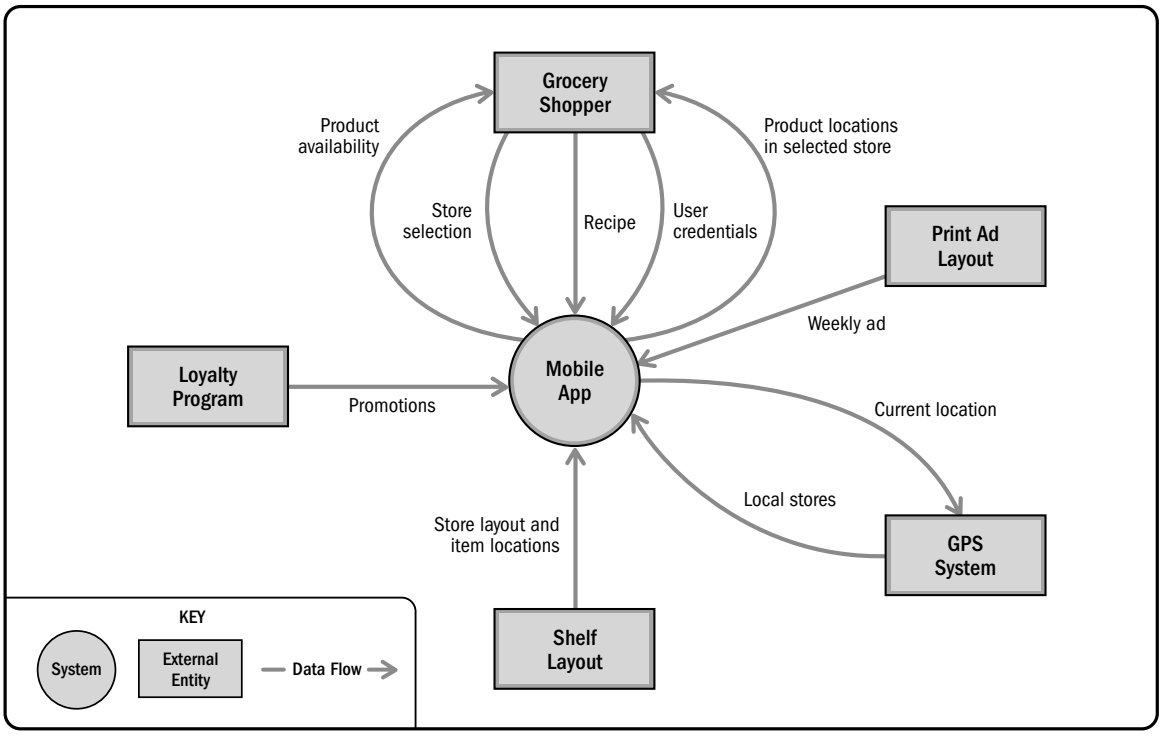


Figure 4-4. Example Context Diagram

4.10.7.4 Feature Model

A feature model is a visual representation of all of the features of a solution arranged in a tree or hierarchical structure. The structure can be horizontal or vertical. A feature is a group of related requirements described in a few words. Most projects have features at varying levels; the top-level features are called Level 1 (L1) features, followed by Level 2 (L2) features, and so on. Most feature models will have three or fewer levels of features. A given branch of the feature model always has a feature at the end of it, with lower level features hanging off the branch.

> *Example*—Figure 4-5 demonstrates features for a simple Recipe Box solution. This feature model shows five L1 features such as list, recipe, and guide with the respective L2 features. For a few of the L2 features, there are L3 features also shown. The different choice of font color in this diagram is used to denote scope. For example, list, recipe, and guide are included in the current iteration and the rest (the features in light gray) are for future iterations. Feature models can be embellished using color or patterns to indicate scope. For example, one color could be used to show what is in scope for the current release and another color could show what is out of scope.

- **Usage.** Feature models are helpful to show how features are grouped together and which features are subfeatures of other ones. Feature models are useful because they can easily display up to 200 features across different levels on a single page, which may represent an entire solution's feature set.
- **Relationship to Requirements.** Feature models usually do not show requirements, but rather sets of requirements (features). Feature models help to determine how to organize requirements for business

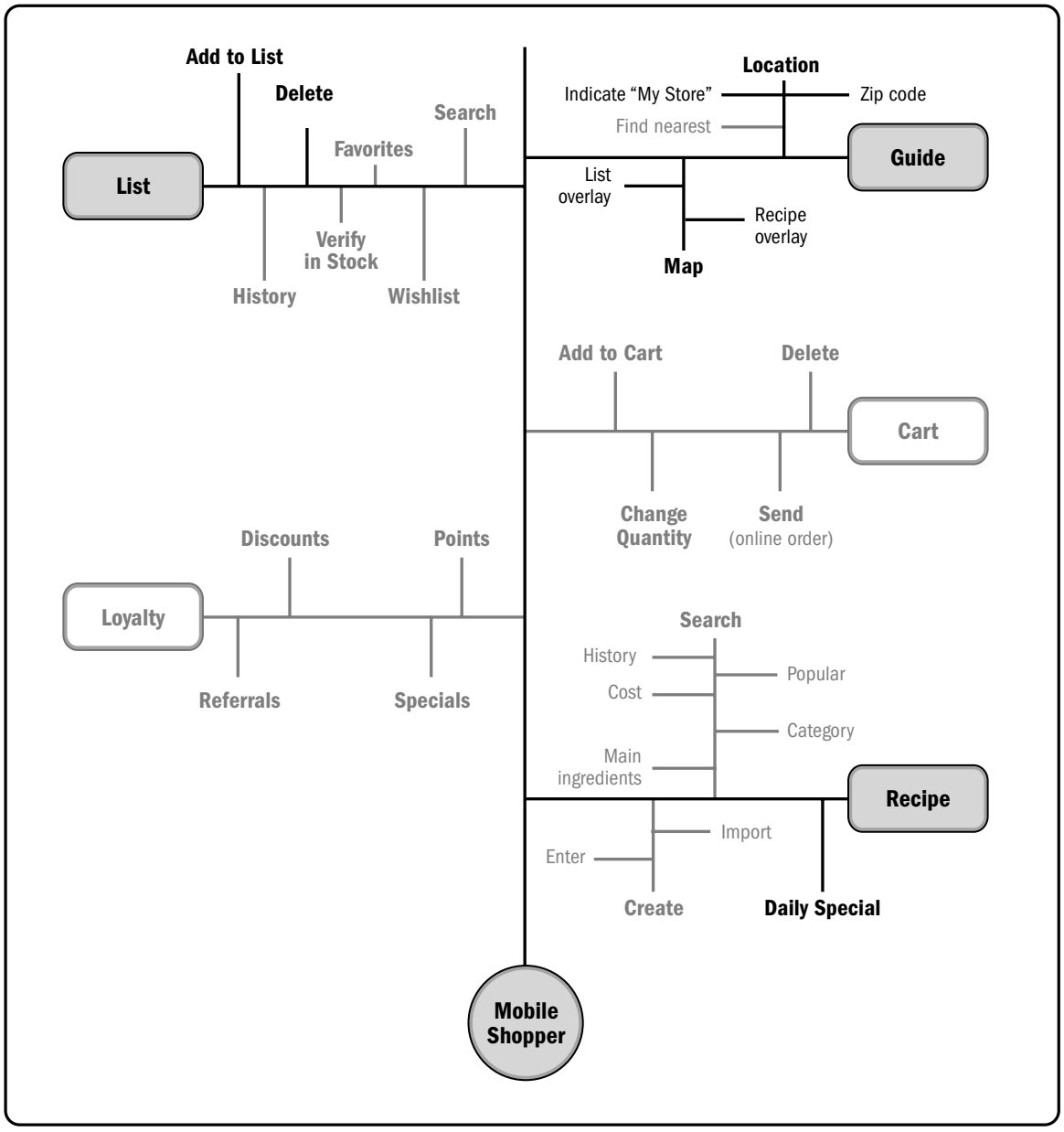


Figure 4-5. Example Feature Model

analysis efforts or lay out features in a requirements document. The features found within a feature model may also be used to trace requirements to ensure that no features or requirements are forgotten.

4.10.7.5 Use Case Diagram

A use case diagram shows all of the in-scope use cases for a system. In a use case diagram, a use case is represented by an oval with the name of the use case within it as shown in the Figure 4-6. An actor is shown as a stick figure. Straight lines in the diagram associate the use cases that the actor interacts with. The association does

not represent the flow of information into or out of the use case. The association merely establishes a connection that shows this actor is in some way associated with the use case. Use cases are further explained in 4.10.8.2 below.

Example—Figure 4-6 shows a use case diagram with a sample set of use cases for the recipe box system.

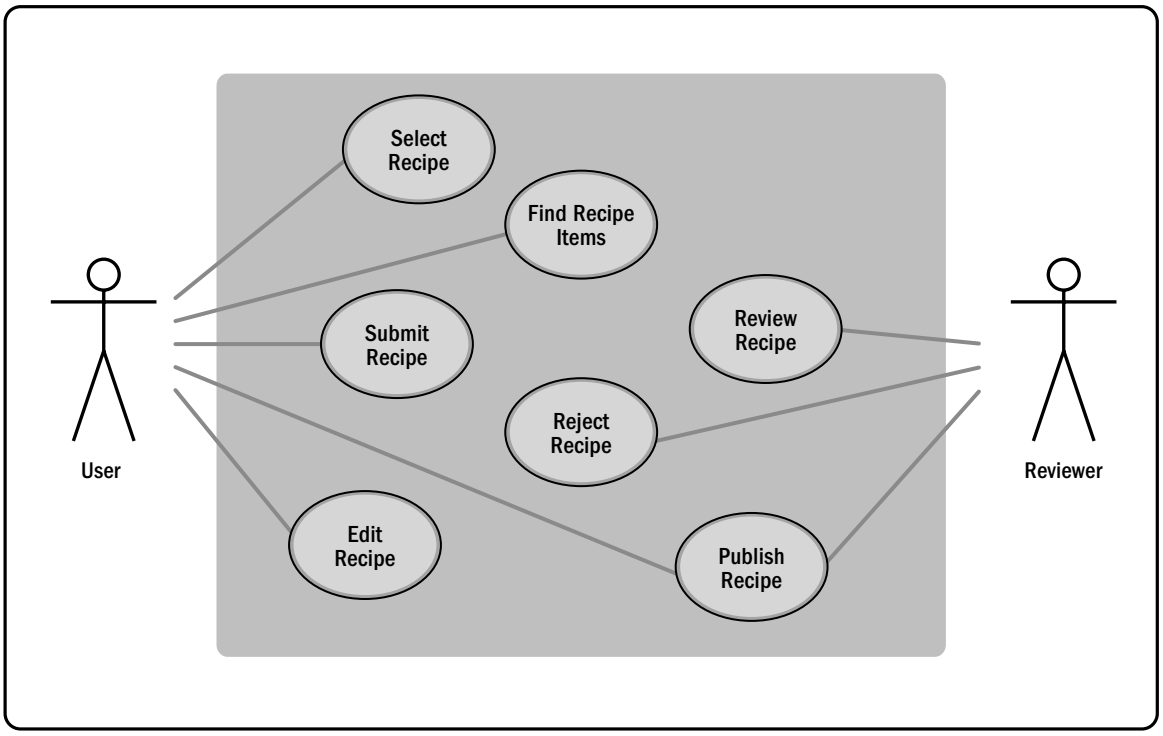


Figure 4-6. Example Use Case Diagram

- **Usage.** Use case diagrams can be used to summarize the scope of a solution, highlighting the main features to be added (i.e., the use cases). These diagrams also show the stakeholders who directly interact with the solution (actors), and the interfaces that need to be created between the system features (use cases) and the actors. Use case diagrams can also indicate use cases that are out of scope, which is helpful to manage stakeholder expectations.
- **Relationship to requirements.** Use case diagrams help the project team plan and track progress in building a solution. These diagrams help to summarize the scope of features and the relationship of features to actors. Use case diagrams do not show requirements, but help to organize requirements for business analysis efforts or layout in a requirements document.

4.10.8 Process Models

Process models describe the user or stakeholder elements of a solution, process, or project.

4.10.8.1 Process Flow

Process flows, also called swimlane diagrams, process maps, process diagrams, or process flow charts, visually depict the tasks that people perform in their jobs. Typically, process flows describe the steps that people take, although they may describe system steps and could be called system flows. In process flows, boxes depict steps, diamonds indicate decision logic, and arrows show the order of flow. Process flows may also contain swimlanes, which group steps together that are performed by the same person, group of people, or system. It is helpful to describe only people or system process steps in a given diagram to reduce shifting the context for the reader between the human and system processes. Process flows are developed to model the as-is processes (e.g., how activities are currently performed) in an organization as well as the to-be processes (e.g., proposed process revisions or new proposed processes).

Example—Figure 4-7 shows the process flow for creating and sending the daily recipe for the recipe box application. This process flow has two swimlanes: the customer and the store representative.

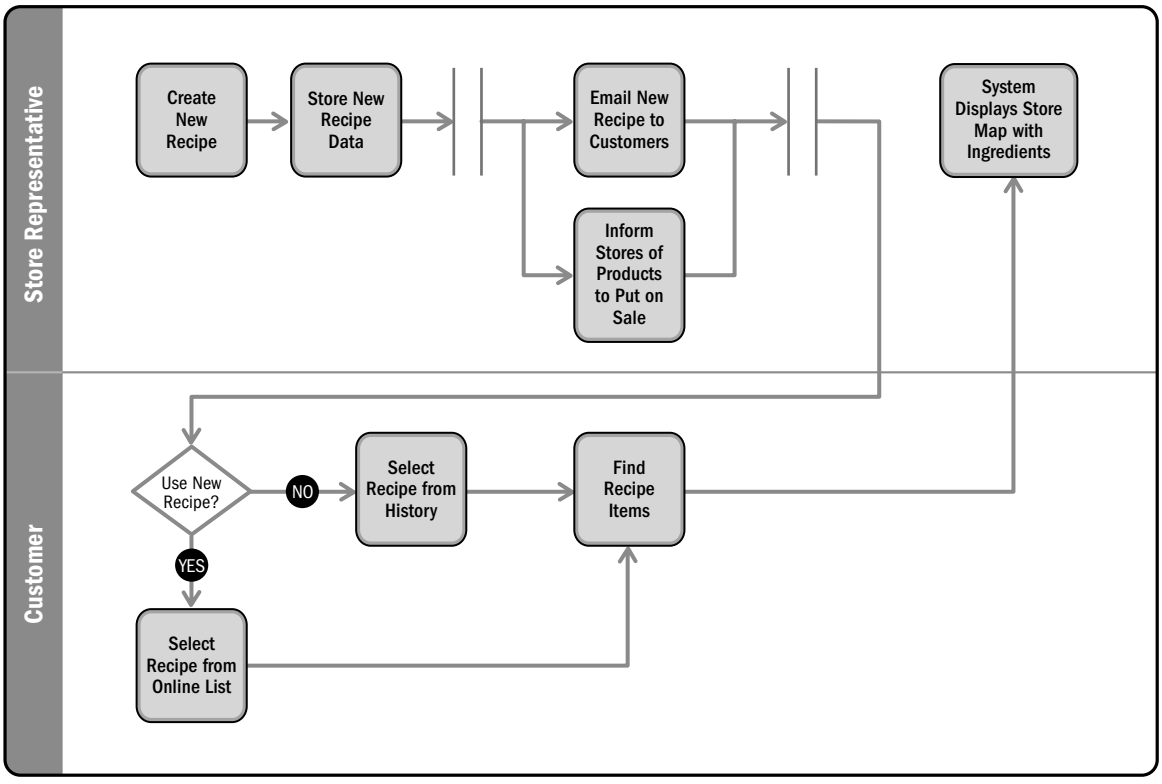


Figure 4-7. Example of Process Flow

- **Usage.** Process flows are valuable on most types of projects where there are people performing tasks in the solution. Process flows are particularly useful for facilitating conversations during elicitation with business stakeholders because process flows are intuitive to create and read. During analysis, process flows are used to identify missing features or requirements by tracking requirements to individual steps

within the process flows. Process flows can be drafted ahead of time or created real-time during sessions and are an easy model for business stakeholders to review as part of requirements verification. Process flows can be used to discuss as-is processes for current solutions and future processes for new solutions to identify changes or gaps. Software developers and testers find process flows to be useful to provide context for a set of requirements.

Process flows are also used to show key performance indicator (KPI) metrics, either the baseline or target metrics. Each KPI is shown in brackets over the step or steps the metric applies to. This is useful when creating solutions to replace existing solutions where there is already a performance level threshold that needs to be maintained.

- **Relationship to requirements.** Process flows are an easy model for deriving requirements by consideration of the functionality or qualities needed to support each step of a business process. This is done during elicitation sessions or during analysis. When requirements are already specified, these can be traced back to the process flow steps to see if there are requirements missing (steps that do not have sufficient requirements mapped) or requirements that are unnecessary (requirements that do not support any steps in a process).

4.10.8.2 Use Case

A use case describes a set of scenarios. A scenario is any single pass through a system to achieve a goal for the primary actor. A use case is a series of activities, actions, and reactions that take the primary actor from initiation to successful completion of the goal. Textual use cases are represented in a standardized document template or in tabular form with standardized columns.

Use cases represent the functional aspects of a system or operation and, as such, are not used to document the nonfunctional aspects of a system (e.g., how fast the product should work, how durable it needs to be, what the capacities will be, etc.). When documenting nonfunctional requirements on a project with use cases or user stories, consider placing the nonfunctional requirements into a separate document. Nonfunctional requirements generally apply to the entire system rather than a single use case; therefore, it makes better sense for them to reside outside of any particular use case.

Common fields include:

- **Name.** A verb phrase that indicates the goal of the use case.
- **Description.** A simple explanation of the use case.
- **Actors.** Roles that are active participants in the use case.
- **Organizational benefit.** Describes why the use case is important to the project or organization; used for prioritization.
- **Trigger.** The event that causes the use case to start.
- **Preconditions.** Describes everything that should be in place prior to the use case starting in order for the use case to succeed.

- **Normal flow.** The normal course of steps to move from the preconditions to the post conditions.
- **Post conditions.** Everything that has changed in the environment at the end of a use case.
- **Alternate flows.** Alternative sets of steps an actor can take to achieve the goal other than what is described in the main flow. These flows are often branch points from steps in the main flow.
- **Exception flows.** Errors or disruptions in the normal flow that require an actor or system to perform a different action to respond to the exception. These are often branch points from steps in the main flow and will usually terminate a use case. Exception flows result in failure or nonachievement of the goal.

A single stakeholder may be represented by multiple roles that mirror the multiple roles that the stakeholder plays in the business. Similarly, many stakeholders may be represented by a single role. The normal flow is commonly referred to as the happy path or main success scenario.

Example—The example shows a use case for a mobile application that a shopper can use to find the ingredients for a recipe in a store. The alternate and exception flows are referenced from various steps in the main flow to show where the flow branches.

- **Usage.** Use cases are used when there are complex back and forth interactions between users and systems. These can be used during elicitation sessions to discover and describe complex interactions. Use cases are used during analysis and then later reviewed with stakeholders. Similar to process flows, use cases offer context for a scenario and specifically show how stakeholders envision the solution. Use cases are helpful as a starting point for creating tests, including user acceptance tests. Use cases can be built and implemented iteratively. A use case diagram is not required when creating use cases, but it is a quick way to visually depict which actors are associated with multiple use cases and what the full scope of a use case is. See Section 4.10.7.5 for further information on use case diagrams.
- **Relationship to requirements.** Use cases typically are not standalone requirements but help to identify functional and nonfunctional requirements. During analysis, each step is analyzed to look for requirements to support the step. In particular, system steps will likely have requirements traced to them. Some organizations may choose to use the list of use cases to serve as the functional requirements bypassing the creation of a separate list of requirements statements. While this may save time, testing efforts are more difficult because traceability is not maintained at the requirement level. A failure of one functional requirement will cause the entire use case to fail. While use cases help to identify nonfunctional requirements, it is often preferred to document nonfunctional requirements external to a particular use case, because these often apply to the whole system.

4.10.8.3 User Story

A user story is a statement, written from the point of view of the user, and describes the functionality needed in a solution. Typically, a user story is used in an adaptive methodology, such as agile development, but can be used in any methodology approach. A user story often takes the format of:

As an <actor>, I want to be able to <function>, so that I can <business reason>.

Table 4-4. Example of Use Case

Name	Find Recipe Items
ID	UC_001
Description	A daily email with the featured recipe is sent to customers who have opted-in to the program. The customer opens the Recipe Box application directly on their iOS or Android device. The customer can choose a past recipe or use the current one. With a recipe chosen, the customer chooses a store or uses a store already indicated a "My Store." A map of the store is displayed with an overlay showing the locations of the items for the recipe.
Actors	User (customer on a mobile app)
Organizational Benefits	Customer visits store to make purchases related to the recipe which are high-margin cooking items.
Triggers	Customer clicks on link in email on mobile device or directly opens application
Preconditions	Recipe Box application opened successfully
Post conditions	Customer views a map of the store with recipe items and locations indicated
Normal Flow	1. System shows short description of the daily recipe 2. Customer selects current recipe or chooses a past recipe 3. Customer chooses to use their local store ("My Store") (see AC1, see AC2) 4. System displays map of the store with the recipe items overlaid as icons 5. Customers can select icons to see item aisle and shelf location (see EX1)
Alternate Flows	AF1—Local store not yet indicated 1. System prompts for zip code for stores 2. Customer enters zip code 3. System lists stores nearby to zip code (see EX2) 4. Customer chooses a store to be "My Store" and it is used for recipe map 5. Return to MC step 4 AF2—Local store not used (whether a "My Store" is chosen or not) 1. System prompts for zip code for stores 2. Customer enters zip code 3. System lists stores nearby to zip code (see EX2) 4. Customer chooses a store from the list 5. Return to MC step 4
Exception Flows	EX1—Selected store does not have an items for the recipe 1. System alerts user about out<>f-stock items 2. Return to MC step 5 EX2—No stores in zip code 1. System alerts customer that there are no stores found in zip code and to enter different zip code 2. Return to AC1 or AC2 step 2

The function provides a small discrete piece of business value or function. The elements of the INVEST acronym can be applied to all requirements, regardless of the format, to ensure quality of the user stories:

- **Independent.** Each story should stand alone, avoiding the creation of dependencies between stories, as much as possible.
- **Negotiable.** The story is subject to negotiation at all times regarding the content, priority, form, and function of the story, and becomes more concrete just before implementation.
- **Valuable.** The story only defines features or functions that are valuable to the business and that help solve the business problem.
- **Estimable.** The story should be clear enough to generate a valid estimate or lead to a discussion that will generate an estimate.

- **Small.** Stories should be small enough to be implemented, adding an increment of real value, within a single iteration.
- **Testable.** Each story should be independently verifiable.

When using user stories, acceptance criteria are provided that are used to confirm that the story is completed and working as expected. When a user story is too large to be completed in a single iteration, it is considered to be an epic. Epics are decomposed further into stories (or additional epics). Stories are used by the development team to build the product.

Example—Table 4-5 shows one epic and three related user stories. The epic describes what the user will get from this product and the user stories describe three specific types of activities the user should be able to do.

Table 4-5. Example of User Story

Epic: As a customer, I want to go to the store and easily buy ingredients needed for a recipe.	
User Story	**Acceptance Criteria**
As a customer, I want to be able to find past recipes so I can prepare them again.	1. Customer can search for past recipes 2. Search terms can be by recipe name, ingredient, or date 3. A search can return 0, 1, or many results 4. For one or many results, a user can choose a recipe from the list
As a customer, I want to select a store to purchase recipe ingredients so I can choose a store location that is close to me.	1. Customer can enter a ZIP Code and get a list of stores in or near that ZIP Code, out to 20 miles from the ZIP Code 2. Customer can select a store from the list of stores 3. If no stores are found, the customer is informed
As a customer, I want to have a store map of where recipe ingredients are so I don't have to hunt for them.	1. Customer has selected a recipe and a store 2. Customer is shown a map of the store with icons representing the recipe ingredients 3. Recipe icons can be selected by click or hover, and will show the ingredient name as well as the aisle and shelf location

- **Usage.** User stories focus on what the user is looking to accomplish and are written from the user's perspective. They can be derived from process flows when the model is used. User stories are helpful on projects, because they are easy models for business stakeholders to engage in creating. In agile methodologies, user stories populate a backlog and are used as a basis for prioritizing future development. As user stories get closer to the top of the backlog, these should be elaborated using relevant modeling techniques to generate enough details for development to occur (known as "grooming the backlog"). Acceptance criteria should be developed at this time. For additional information on creating acceptance criteria, see Section 6 on Solution Evaluation.
- **Relationship to requirements.** A user story contains many requirements; therefore, it serves as a functional grouping of requirements. Stories can be traced directly to business objectives to substantiate the value of the requirements and can also be traced to elements in other models. User stories can be used to manage, prioritize, trace, and allocate functionality to releases and iterations. Although user stories are not detailed, they contain acceptance criteria and express user needs.

4.10.9 Rule Models

Rule models help to identify and document the business rules, business policies, and decision frameworks that need to be supported by the solution. Business rules are constraints about how the organization wants to operate and are usually enforced by data and/or processes and tend to be true over time. When analyzing business rules, the objective is to capture what should or should not be allowed in a business enterprise. An important aspect of analyzing business rules is identifying the source of a rule. A key element of business rules analysis is the absence of technology.

4.10.9.1 Business Rules Catalog

A business rules catalog is a table of business rules and related attributes. Common attributes to capture in a business rules catalog include a unique ID for the business rule, the rule description, and the type of business rule (there are multiple types of business rule classifications that can be followed), and references to other related documents. Business rules themselves are not processes or procedures, but rather describe how to constrain or support a behavior. Business rules apply across processes and procedures and guide the organization's activities. These rules should be written in plain language and each row should describe one rule. Business rules should not be nested and each should be able to stand independently. When creating business rules, the rules need to be correct, verifiable, and consistent. The catalog can be used to refer to related requirements or governance documents.

Example—The partial business rules catalog in Table 4-6 shows a few business rules for the recipe box. For example, only customers who have opted-in for emails will receive recipe box emails and that no personally identifiable information is allowed to be in the emails.

Table 4-6. Example of Business Rules Catalog

Recipe Box				
BR ID	**Business Rule Title**	**Business Rule Description**	**Type (fact, computation, constraint, other)**	**References**
BR01	Recipe Email Opt-in	Recipe emails will only be sent to customers who have opted-in and have a valid email address.	Constraint	See corporate email policy
BR02	No PII in Recipe Email	Recipe emails will not contain any personally identifiable information (PII).	Constraint	See corporate email policy
BR03	Ingredients in stock	A new recipe will not be sent when more than 10% of the stores have a restocking status of greater than 24 hours for any of the ingredients	Computation	Will use inventory reporting system

- **Usage.** Business rules should be maintained in a repository such as the business rules catalog. The rules need to be kept current and should be updated in the same manner as any other analysis model. The business rules catalog can be maintained at a level above an individual project, because business rules are often not specific to one project. Business rules catalogs can be traced to business objectives and other requirements types or analysis models in a traceability matrix to ensure that all business rules are captured. For example, decision trees help to identify business rules, so a mapping of business rules to decision trees would help to ensure completeness in this catalog.

- **Relationship to requirements.** It is common for business rules to be identified in the normal course of eliciting and analyzing requirements. These rules could lead to functional requirements that are necessary to support the business rules. Rules also emerge in discussions of functional requirements that involve decisions.

4.10.9.2 Decision Tree and Decision Table

Decision trees and decision tables depict a series of decisions and the outcomes they lead to. Decision trees and tables are often used to model business rules. Decision trees work best with binary choices (i.e., yes or no), and decision tables can be used when more choices exist and the analysis is becoming complex.

Decision trees are described in a tree of decision points where each branch represents a different choice. The far right of a decision tree (the leaves) represent the outcomes for a decision or series of decisions. Decision points in a decision tree are commonly represented as text at the branch points or in diamond shapes at the branch points. Decision outcomes are typically shown in boxes. Decision trees are drawn horizontally or vertically (with outcomes at the bottom).

Decision tables use a tabular format where the upper rows in the table represent the decision points and the bottom rows in the table represent the outcomes. Each decision point row enumerates a set of all valid choice combinations. A dash in the cell indicates that the choice does not matter in determining the outcome in that column. Each outcome row is marked to indicate which choice combination is valid. Each column of the table is a business rule that describes the set of choices for each decision point and the outcomes it leads to. When an outcome is unachievable, it is considered indifferent in a decision table.

The decision table consists of four areas as follows:

- **Condition stub.** Lists all of the possible conditions in the process or activity being analyzed.
- **Conditions.** Indicate which conditions are met.
- **Action stub.** Defines the possible actions as a result of the process or activity being analyzed.
- **Actions.** Indicate which actions are taken as a result of the conditions that are met.

  Example—Figure 4-8 shows a sample decision tree for selecting a recipe in the mobile application. For example, if there is a store in range and if there is inventory for the items the shopper wants, and if the locations of the items are available, then the map can display item locations. Decision trees can be arrayed in different forms; below is one approach. Figure 4-9 represents the decision table for this decision logic.
- **Usage.** Both decision trees and decision tables are used to model complex branching logic. During elicitation or analysis, when a business analyst uncovers a series of "if this, then that" statements, either type of decision model is appropriate to use. Decision trees are helpful to identify ways to reduce complex decision logic by looking for redundancies in the structure. Decision tables are useful to ensure all possible combinations of decision choices are considered.
- **Relationship to requirements.** Decision trees and decision tables are used to identify and represent business rules. Both models stand alone as representations of the business rules because they can be implemented directly, as a process or in development. These models can also help to identify any requirements related to supporting those business rules or the specific outcomes.

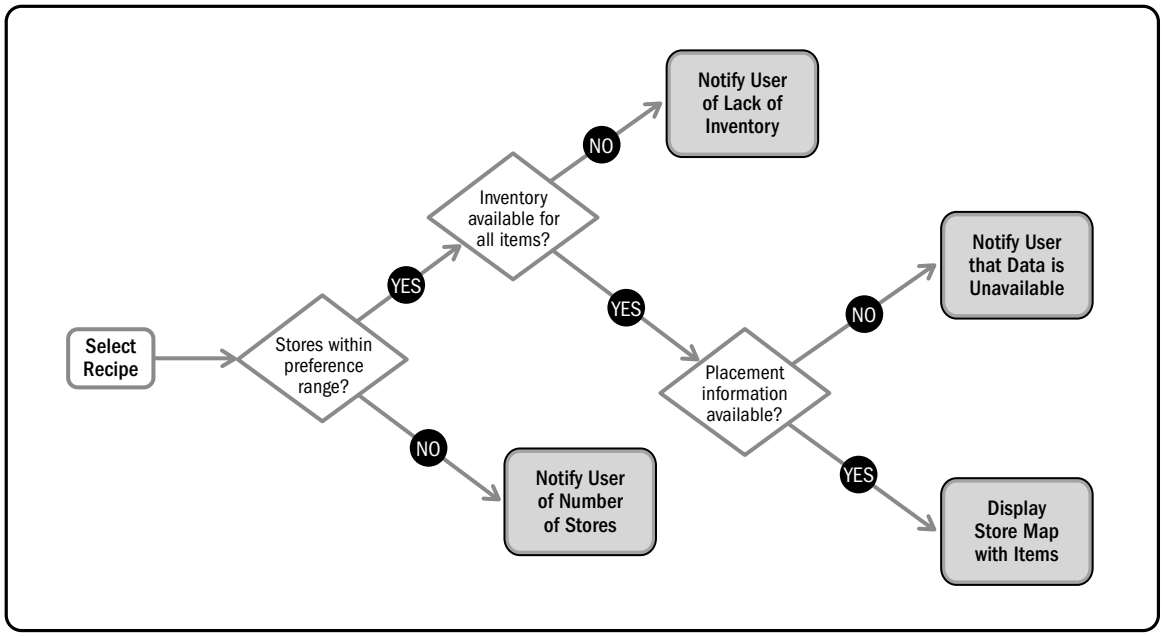


Figure 4-8. Example of Decision Tree

Map Store Decision Table	Rule 1	Rule 2	Rule 3	Rule 4
Conditions				
Stores within preference range	N	Y	Y	Y
Inventory available for all items	–	N	Y	Y
Placement information available	–	–	N	Y
Outcomes				
Notify user when NO stores within preference range	X	–	–	–
Notify user of lack of inventory	–	X	–	–
Notify user that data is unavailable	–	–	X	–
Display store map with items	–	–	–	X

Figure 4-9. Example of Decision Table

4.10.10 Data Models

Data models document the data used in a process or system and its life cycle. These models are used to depict relationships between data, to show how data is related to processes, and to further help extract requirements and related business rules.

4.10.10.1 Entity Relationship Diagram

The entity relationship diagram (ERD), also called a business data diagram, shows the business data objects or pieces of information of interest in a project and the cardinality relationship between those objects. Business data objects are the conceptual pieces of data that the business thinks and cares about and are not intended to refer to exact data objects in a database. Business data objects represent the people, places, things, and concepts that the business cares about. Business data objects are shown as boxes, relationships as lines, and cardinality as labels on the lines. Multiplicity is indicated on the relationship line to represent the number of times (cardinality) that one entity occurs in relationship to the other entity in the relationship, and whether the relationship is required or optional. Cardinality is modeled in several ways, such as crow's foot notation or the 0 (none), 1 (single), and *n* (many) notation as shown in Figure 4-10. The method chosen is usually based on what the organization or business analyst is familiar with.

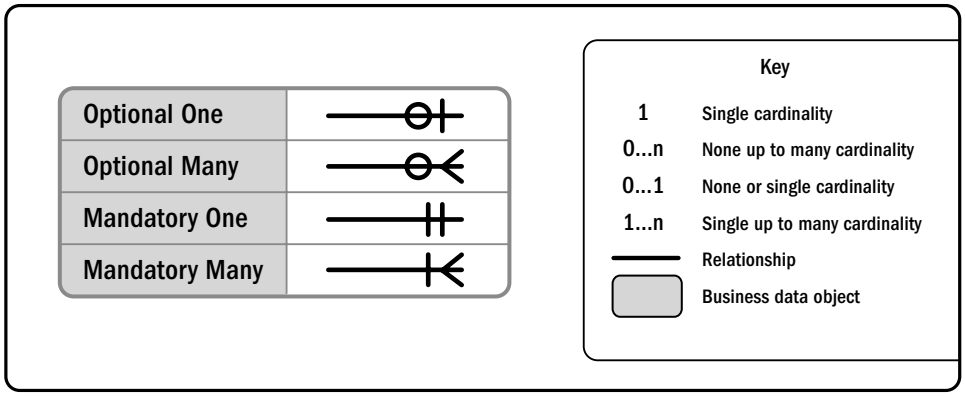


Figure 4-10. Example of Crows' Foot and 1 to N notation

Some ERDs show the nature of the relationship between the objects as a text label on the line.

Example—In Figure 4-11, the entity relationship diagram shows four different objects and their relationships. For example, the customer may have zero or more recipes and will be able to obtain information from one store for a given recipe. The item location is specific to the store and would be captured as an attribute (in a data dictionary), but it is helpful to note that on this type of diagram.

- **Usage.** The entity relationship diagram is a cornerstone model for a project that has a data management component, because it helps with identifying the data that is created in, consumed by, or output from the system. This model is used to define the business data objects and their relationships to one another. The specific attributes of these objects are specified in a data dictionary.
- **Relationship to requirements.** Systems typically manipulate business data objects through functions to allow business data objects within an entity relationship diagram to be traced directly to requirements for these functions. The data objects can be traced to data flow diagrams, ecosystem maps, data dictionaries, and state transition models.

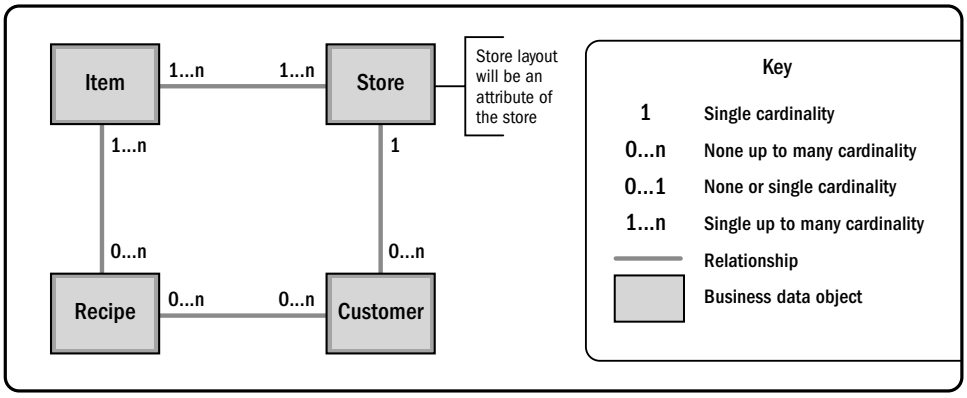


Figure 4-11. Example of Entity Relationship Diagram

4.10.10.2 Data Flow Diagrams

A data flow diagram illustrates the relationships between systems, actors, and the data that is exchanged and manipulated over the course of one or many processes. It is a model that can be used after business data diagrams, process flows, and an ecosystem map have been created. In Figure 4-12, the data stores (shown as 2 parallel lines) show where information is conceptually stored, and the process steps (shown as circles) indicate which functions manipulate or use the data, and which external entities (shown as boxes) create data or consume data.

Example—The data flow diagram in Figure 4-12 shows that the customer selects a recipe and store, which are used in processes to map the store and overlay the ingredients on the map.

- **Usage.** A data flow diagram can be used to describe the movement of data between actors and systems over the course of a process or several processes. Data flow diagrams identify data inputs and outputs for processes, but do not specify the timing or sequence of operations.
- **Relationship to requirements.** Data flow diagrams relate to requirements through the business data objects and processes. While requirements can be traced to the model, it is better used as a tool to help stakeholders and developers understand how data flows through the systems, which then leads to identifying specific data requirements.

4.10.10.3 Data Dictionary

A data dictionary is a tabular format and shows data fields and attributes of those fields. Common attributes include name, description, size, and validation rules. However, any relevant attributes can be captured in a data dictionary.

Example—Figure 4-13 is part of a data dictionary and shows a few attributes for the recipe, store, and items. Business rules may be described in a data dictionary. For example, this data dictionary shows StoreNum is a 3 digit number that is required to be postive and one of the valid store numbers.

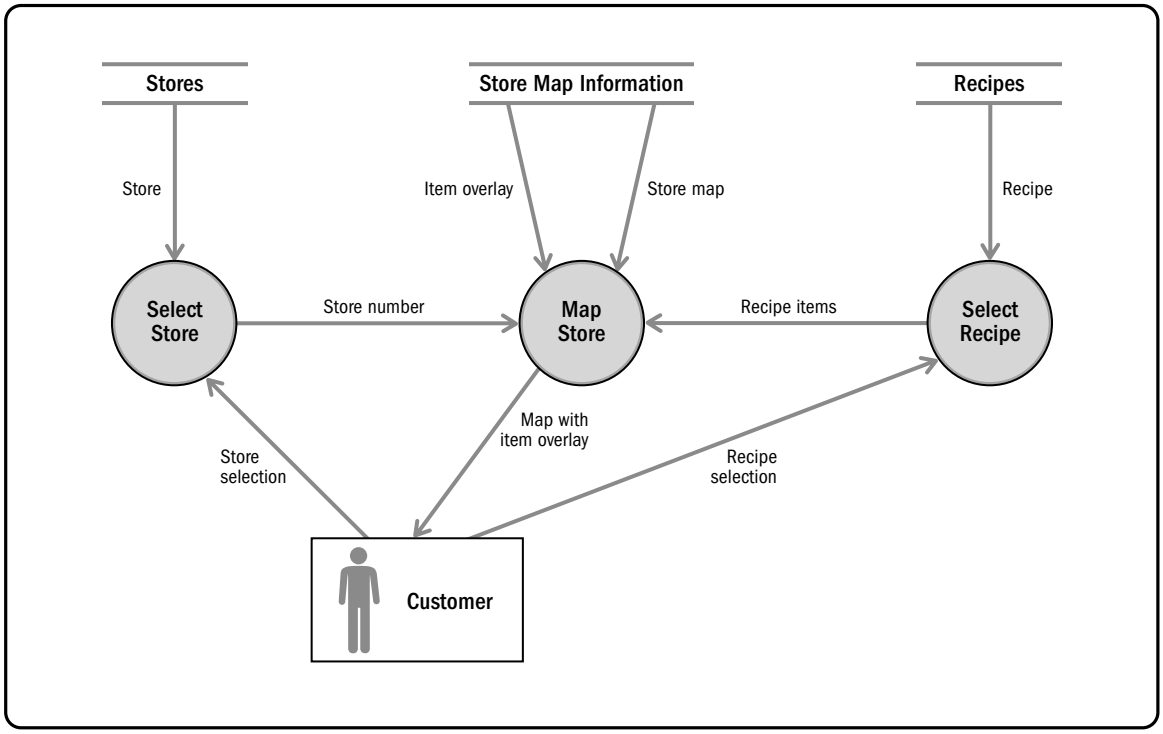


Figure 4-12. Example of Data Flow Diagram

ID	Business Data Object	Field Name	Description	Unique Values?	Data Type	Length	Valid Values
RM01	Recipe	Recipe text	Recipe in formatted layout	Y	Alphanumeric	< 1,000 characters	Structured text (see REC_FRM01)
RM03	Items	Recipe UPC	UPC collection for a given recipe	Y	Alphanumeric	< 1,000 characters	Structured text (see REC_FRM02)
RM02	Store	Store number	Store number for map and overlay	Y	Integer	3 numeric characters	Integer > 0 and valid store number
RM04	Store	Store map	Graphic of store layout	Y	Graphic	640 x 480 pixels	n/a
RM06	Store	Item overlay	Generated array of recipe items, locations, scaled to store layout graphic	Y	Alphanumeric	< 1,000 characters	Structured text (see LAYOUT_FRM01)

Figure 4-13. Example of Data Dictionary

- **Usage.** Data dictionaries are used to specify very detailed aspects of data and to capture data fields and attributes from the business stakeholder's perspective. The information captured does not need to explicitly reflect a database design. However, database designers use data dictionaries as an input to create the database architecture.

- **Relationship to requirements.** Data dictionaries are used to capture very detailed requirements and their business rules. This model can stand alone and does not need redundant requirements statements.

4.10.10.4 State Table and State Diagram

State tables and state diagrams model the valid states of an object and any allowed transitions between those states. State tables are in a tabular format with all of the valid states in the first column and across the first row. Each cell represents the transition from the state in the row to the state in the column. Transitions that are not allowed have cells that are marked with "X," "N/A" or "no." Allowed transitions are represented in cells with either "yes" or a description of the transition event that leads to the transition. State diagrams show exactly the same information as state tables, but it is easier to visualize the valid states and transitions by showing only the allowed transitions. Ovals are used for states, and lines with arrows show the transitions and transition events between states. Some state diagrams are drawn by showing an initiation state and a termination state.

Example—Figure 4-14 is an example of a state table for the states of a recipe, and Figure 4-15 is the state diagram for the exact same scenario. Notice that the states and transitions are the same in both figures; either model may be used to show the same information. Both models show that a reviewed recipe can move to either a rejected or published state. The state diagram makes it easier to see how a recipe moves from drafted to reviewed to published states.

		Target State				
		Drafted	**Reviewed**	**Rejected**	**Published**	**Expired**
Initial State	**Drafted**	X	Reviewer reviews	X	X	X
	Reviewed	X	X	Reviewer rejects draft	Reviewer approves	X
	Rejected	Resubmit as draft	X	X	X	X
	Published	X	X	X	X	120 days elapse with no use
	Expired	Author edits expired recipe	X	X	X	X

Figure 4-14. Example of State Table

- **Usage.** State tables and state diagrams help business analysts specify the life cycle of an object in the solution. For example, objects that go through workflow (e.g., an approval process) are aided by using state models. State tables are useful to ensure that state transitions are not missed, because every possible transition is represented by a cell in the table; when every cell in the life cycle is considered, no transitions can be forgotten. It is more difficult to ensure that state diagrams are complete, but much easier to quickly visualize the life cycle of an object.
- **Relationship to requirements.** State table and state diagrams are models that stand alone and do not require additional requirements statements to be developed and tested correctly. The exception to this is

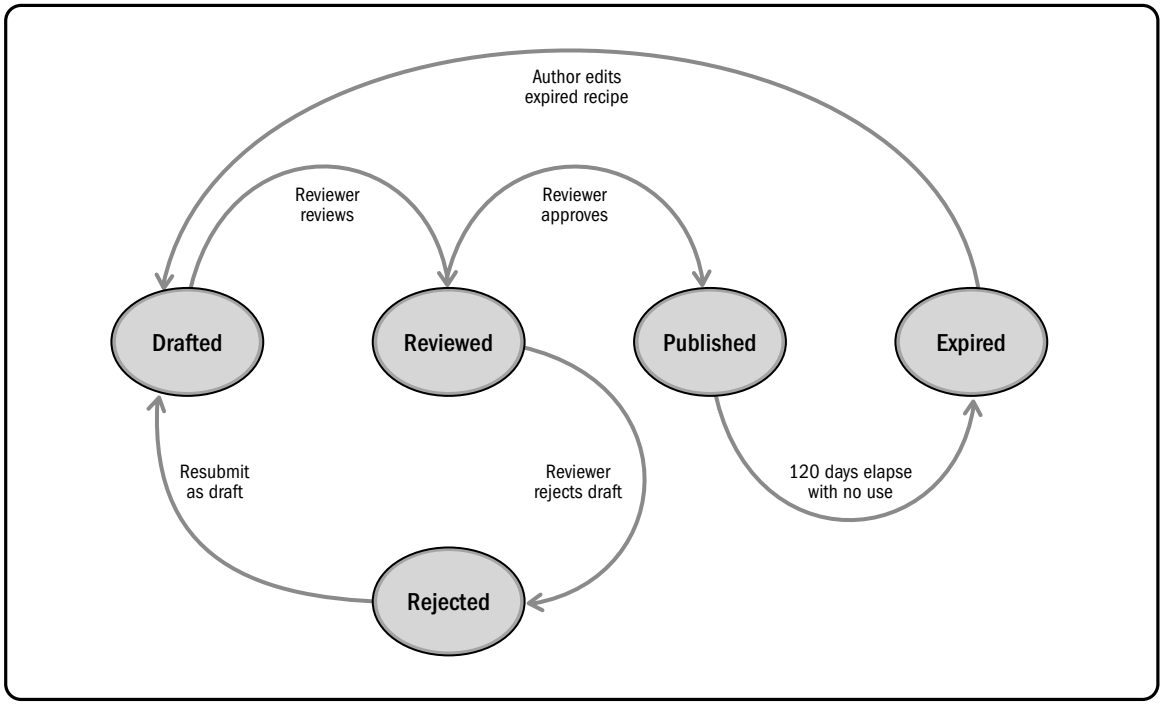


Figure 4-15. Example of State Diagram

when the transition events are too complicated to fit in the cell or on the arrow, in which case, additional detail is included outside of the model. State tables are also used to confirm or find gaps in data and processes that have been specified by other models and are used often to model business rules.

4.10.11 Interface Models

Interface models depict the relationships within a solution to gain an understanding of which interfaces there are and the details of those interfaces.

4.10.11.1 Report Table

A report table is a model that captures the detailed level requirements for a single report. Common attributes of a report include: name, description, decisions made from the report, objectives, audience, trigger, data fields, data volume, frequency, display format, and calculations. These attributes should be specified alongside a prototype or example of the actual report, when possible, because it adds context for the textual information in the report table. While it is not necessary to complete all fields, the fields can be used to help identify what to think about when eliciting reporting requirements.

Example—Figure 4-16 shows a prototype of the Mobile App Usage Report and Figures 4-17 and 4-18 show the report table model (the model is split into two figures here for sizing but is typically one table). Fig 4-16 provides data for each store on the number of transactions, number of products per transaction,

Mobile App Usage Report (MAUR)
Reporting Period: September 07, 2014–September 13, 2014

Number of App Downloads	Number of Recipe Uploads	Store ID	Region	Number of Loyalty Program Transactions	Number of Loyalty Program Transactions by Users with Recipe Uploads	Average Number of Products per Transaction for Loyalty with Recipe Transactions	Average Number of Products per Transaction for Loyalty without Recipe Transactions
10,732	6,907	0	0	78,563	4,576	34	10
		23548	Central Texas	10,723	2,469	32	17
		54721	Central Texas	6,093	1,098	33	15
		56098	Central Texas	15,497	543	30	12

Figure 4-16. Example of Report Prototype

	Element	Description
Top-Level Elements	Unique ID	REP_MBL001
	Name	Mobile App Usage Report (MAUR)
	Description	The MAUR contains the total numbers of new app downloads, recipe uploads, and frequency of visits by app users
	Decisions Made from Report	The MAUR will be used to determine if the Recipe Box feature is being utilized by mobile app users and if users are actually purchasing the products in stores
	Objective	Business Objective 01 (improve profit per visit)
	Priority	1 of 16
	Functional Area	Sales and Marketing
	Related Reports	Inventory Report (REP_INV001), Sales Report (REP_REV001)
	Report Owner	Marketing Manager
	Report Users	VP of Sales, VP of Marketing, Sales Representatives, Marketing Department, Mobile Services Representatives
	Trigger	MAUR is generated automatically on Monday morning at 2 a.m.
	Frequency	Weekly
	Latency	MAUR is delivered within 2 minutes of being triggered and contains data for the previous week (Monday 12:00 a.m.–Sunday 11:59 p.m.)
	Transaction Volume	Approximately 100,000 transactions are logged weekly
	Security	Viewable by all sales and marketing team members
	Persistence	All settings are saved between report executions
	Visual Format	The Reporting period will be noted above the MAUR Matrix in the form "Month DD, YYYY–Month DD, YYYY" MAUR is a matrix with the following columns: Number of App Downloads, Number of Recipe Uploads, Store ID, Region, Number of Loyalty Program Transactions, Number of Loyalty Program Transactions by Users with Recipe Uploads, Average Number of Products per Transaction for Loyalty with Recipe Transactions, Average Number of Products per Transaction for Loyalty without Recipe Transactions The first row contains metrics for all stores The Store ID and Region will be "0" for the first row
	Delivery Format	The report is emailed in an Excel file
	Interactivity	The MAUR retains normal interactivity available in Excel files
	Drilldowns	N/A

Figure 4-17. Example of Top-Level Elements in a Report Table

	Element	Description
Field Elements	Filtered By	All columns can be filtered The default setting displays all data
	Grouped By	Rows are grouped by Region
	Sorted By	All columns can be sorted The default setting is to sort by Store ID within Region
	User Input Parameters	N/A
	Group Calculation	N/A
	Calculated Fields	Each calculation is done for the duration of the report period specified: App.DownloadTotal = sum of app downloads Recipe.UploadTotal = sum of recipe uploads Transactions.LoyaltyTotal = sum of transactions using a Loyalty Program ID Transactions.LoyaltyRecipeTotal = sum of transactions using a Loyalty Program ID with recipe upload(s) Transactions.AverageProductsPerLRT = total number of products purchased in Transactions.LoyaltyRecipeTotal divided by Transactions.LoyaltyRecipeTotal Transactions.AverageProductsPerLNRT = total number of products purchased in Transactions.LoyaltyNoRecipeTotal divided by Transactions.LoyaltyNoRecipeTotal
	Displayed Fields	All fields are shown with rounding to the nearest integer App.DownloadTotal Recipe.UploadTotal Store.StoreNum Store.Region Transactions.LoyaltyTotal Transactions.LoyaltyRecipeTotal Transactions.AverageProductsPerLRT Transactions.AverageProductsPerLNRT

Figure 4-18. Example of Field Elements in a Report Table

new mobile application downloads, and recipe uploads. This information allows the sales and marketing department to establish a correlation between the mobile app and recipe feature and the actual changes in the average number of products purchased in each transaction. There are many different types of report table templates. This is one example and provides one sample of the type of information that could be placed in a report table model.

- **Usage.** Report tables are straightforward, can be created for each report, and are helpful to provide additional details about reports that cannot be gleaned by looking at a mockup. Using a report table with attributes allows the business analyst to specify the type of information to be included in the reports, thereby ensuring that details are not forgotten or overlooked in the solution. When report data has multiple sources, the business analyst needs to define the system of record. The answer may require either a business or a technical decision. When mockup tools are available, teams may choose to use the tool instead of using the table. The business analyst should make sure the tool provides all of the same opportunities to capture requirements as are provided by the report table. Missing these opportunities can result in missed requirements.
- **Relationship to requirements.** The information in a report table model represents the actual report requirements; therefore, no additional requirements are necessary. Stakeholders can use the report table and a mockup of the report to fully understand the report requirements.

4.10.11.2 System Interface Table

A system interface table is a model of attributes that captures all of the detailed level requirements for a single system interface. The system interface table is in a tabular format and typically includes attributes such as source system, target system, volume of data passed, security or other rules, and the actual data objects passed.

Example—Table 4-7 is an example of a system interface table for the interface between the grocery store and the customer's mobile device. This system interface table specifies that information about the store and recipes are synced daily from the store to the mobile device.

Table 4-7. Example of System Interface Table

System Interface			
Source	Store		
Target	Customer mobile device		
ID	MS_01		
Description	Passes store and recipe information to mobile application		
Frequency	Daily		
Volume	<1000 recipes		
Security Constraints	None		
Error Handling	See Sync_Store_Recipe_System_Flow		
Interface Objects			
Object	**Field**	**Data Dictionary ID**	**Validation Rule**
Store	StoreNum	RM02	Positive integer
Store	StoreMap	RM04	JPEG image
Recipe	RecipeText	RM01	Well-formed structured text (ref data dictionary)
Store	ItemOverlay	RM06	Well-formed structured text (ref data dictionary)

- **Usage.** System interface tables are used to specify the details for each interface between the systems in the solution. Typically these tables specify requirements for both the source and/or target system, but it could be only one of these. The attributes of a system interface table ensure that details about the interface are not forgotten.
- **Relationship to requirements.** System interface tables are detailed enough to represent the actual interface requirements and do not need to have other requirements written.

4.10.11.3 User Interface Flow

A user interface flow displays specific pages or screens within a functional design and plots out how to navigate the screens according to various triggers. The boxes in this diagram are the main screens in the user interface. The lines show the flows allowed between screens.

Example—Figure 4-19 is a user interface flow that shows the transition between the screens in the mobile application. For example, a user can move from the login screen to the register screen to the search recipe screen, or directly from the login screen to the search screen.

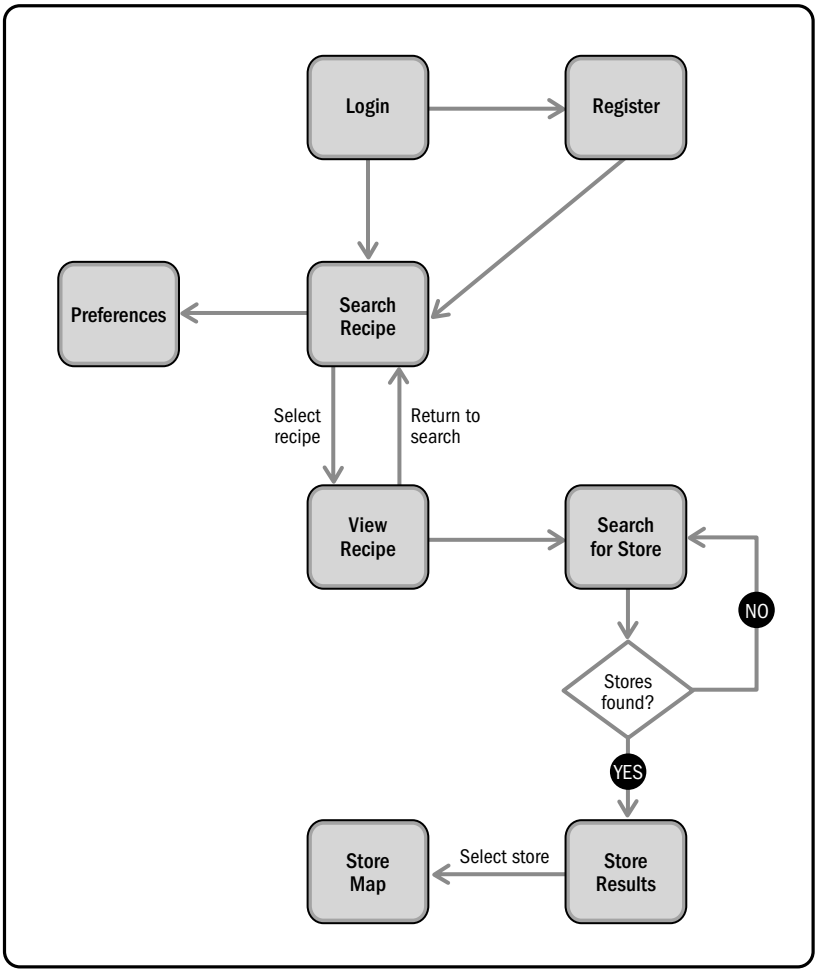


Figure 4-19. Example of User Interface Flow

- **Usage.** Typically, user interface flows are used in the solution definition stage of a project and help track all of the screens that need to be further defined. This model is applicable only when there is a user interface as part of a solution. Interface flows can be used during elicitation sessions to determine more details about the functions that take users between the screens.
- **Relationship to requirements.** This model does not reflect individual requirements statements. The user interface flow can be traced to other requirements models such as the display-action-response models, process flows, and individual requirements.

4.10.11.4 Wireframes and Display-Action-Response

The display-action-response model is used in conjunction with wireframes or screen mockups to identify page elements and the functions, if any, they are attached to. Each wireframe is broken down into user interface elements, which are then described from a display perspective and a behavior perspective. A user interface (UI) element table is created for every element on the screen that has user interface requirements. Each table describes

the user interface element's display requirements under different preconditions and behavior requirements under different preconditions and user actions.

While this type of user interface analysis is sometimes performed by user experience analysts or human factors experts, the business analyst is often called upon to perform this function. In general, user interface analysis focuses on profiles of the users who interact with the system, and precedes interface design. The business analyst, working in conjunction with the user experience analyst when one is assigned, analyzes the user interfaces to see how well these interfaces meet the general principles of human-machine interface:

- **Compatibility.** Minimize the amount of information recoding that will be necessary for operators/users (Write Once, Read Anywhere).
- **Consistency.** Minimize the difference in dialog both within and across various human computer interfaces; follow a user interface standard such as common user access (CUA).
- **Memory.** Minimize the amount of information that users need to maintain in short-term memory (use reference tables or default information to prefill forms).
- **Structure.** Show users the structure of the system so that they are able to navigate through the interface intuitively (effective grouping of similar data elements).
- **Feedback.** Provide users with feedback and error correction capabilities (something happens on the interface every half second).
- **Workload.** Keep user mental workload within acceptable limits.
- **Individualization.** Allow customization of the interface to determine the requirements for individualization.

  Collaboration Point—Because the information elicited is complementary, the user interface analyst and business analyst should work together to support each other when eliciting requirements.

  Example—Figure 4-20 shows a simple wireframe mockup of the login screen. Figure 4-21 shows one display-action-response model for the screen, specifically the display and behavior requirements for the password button. This model specifies requirements to the level of what gets displayed when the user types their password (privacy dots) or what happens when the user selects a field (keyboard appears).

- **Usage.** This model is helpful when there is a user interface for a solution. The display-action response model is typically used when precision is needed for detailing the display and interactions in a user interface. This could occur when defining user interfaces for a large number of users, high-risk user interfaces, or when a development team is unfamiliar with the product. These models are helpful for specifying user interface requirements because the model places the individual requirements statements in the context of the elements on the screen.
- **Relationship to requirements.** Display-action response models trace directly to wireframes, user stories, user interface flows, and data dictionaries. These models contain user interface requirements and do not need further specification.

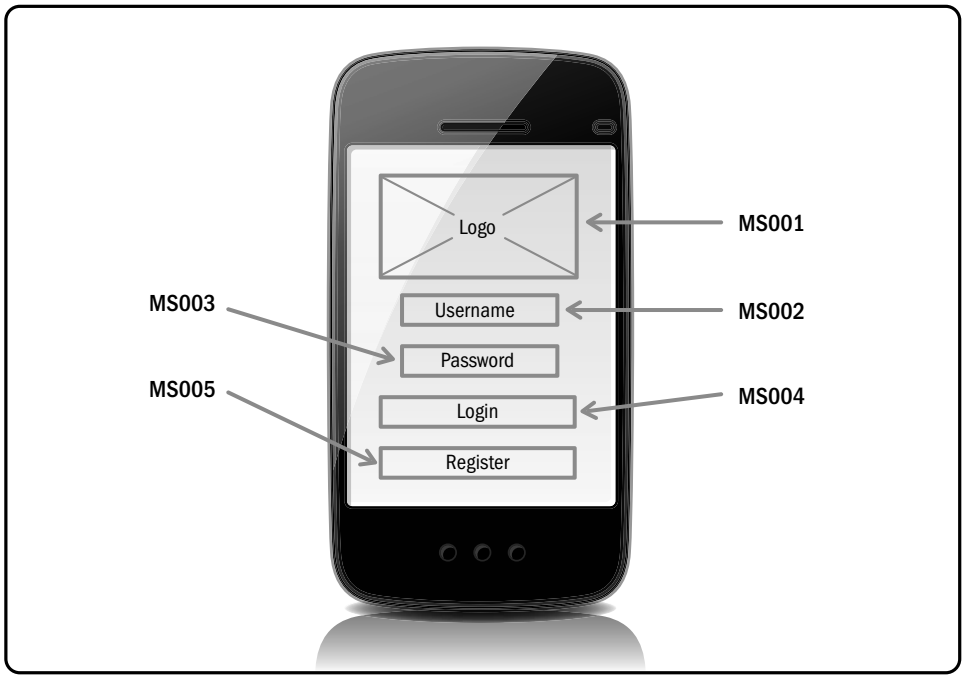


Figure 4-20. Example of Wireframe

UI Element: Password Field		
UI Element Description		
ID	MS003	
Description	A field for the user to enter their password	
UI Element Displays		
Precondition	**Display**	
On login screen	Display field	
No text entered	Grayed out password	
Text entered	Privacy dots are displayed per character	
UI Element Behaviors		
Precondition	**Action**	**Response**
On login screen	Select field	Keyboard appears
Text entered	Enter additional text	Privacy dots per character appear

Figure 4-21. Example of Display-Action Response Model

4.11 Document the Solution Requirements

After analyzing all of the information that has been elicited, the business analyst documents the resulting requirements in one of many forms, depending on the organization, the project needs, and the project life cycle being used. Regardless of the form the requirements take, when packaged together, a set of requirements defines the solution scope to the business problem or opportunity. The business analyst prepares the requirements package so that the solution team understands how to develop the solution. Documentation can be produced in various levels of formality and in many forms, and is often dependent on the selected project life cycle. The solution may not be the complete solution, as in the case of a project following an adaptive project life cycle, but it should represent the solution based on the information available at that point in time.

4.11.1 Why Documentation is Important

Documented requirements serve a multitude of purposes, such as the following:

- Baseline to validate the stakeholder needs;
- Baseline definition of the solution for the business problem or opportunity;
- Primary input to the design team, the developers, testers, and quality assurance;
- Basis for user manuals and other documentation whether written or embedded;
- Supporting detail for contractual agreements, when applicable (e.g., the requirements are a core input to a statement of work (SOW) for a request for proposal (RFP));
- Starting point for the evolution of the solution;
- Foundation for reusability by other project teams who need an understanding of the project details while it is in process or after implementation; and
- Baseline for an audit of regulated industries and high-risk projects that are required to have documented requirements.

Despite the importance of documenting the requirements, keep some factors in mind about requirements documentation:

- Requirements documentation is only one of several techniques to ensure consensus among all of the stakeholders as to the behavior of the solution.
- Documentation should not replace communication and collaboration.

4.11.2 Business Requirements Document

Business requirements are the goals, objectives, and higher-level needs of the organization and provide the rationale for a new project. Business requirements recognize what is critical to the business and why it is critical before defining a solution.

In some organizations, a business requirement is considered to be the high-level requirement, for which user or stakeholder requirements are then used to document the solution. Other organizations use the term business requirements to refer to any requirement that is not a system or technical requirement. Business requirements may be assembled in a business requirements specification or may be part of a larger document that contains all of the requirements. Organizations that use spreadsheets to capture requirements may use a hierarchical structure, with the business requirement as the starting point. In a requirements management tool, business requirements are often grouped by assigning a category or attribute.

4.11.3 The Solution Documentation

Solution documentation is the documentation that is comprised of the features, functions, and characteristics of the product or service that will be built to meet the business and stakeholder requirements. The work of the solution development team is heavily dependent on the solution documentation, because it serves as the blueprint for the product that the solution team is being asked to build. When development work is outsourced, it is essential for the solution documentation to be precise and detailed, because the outsourced team often lacks the business knowledge that an internal development team has. The business stakeholders have a role to review, validate, and approve the solution documentation. The level of formality for these processes is dependent on the selected project life cycle.

The solution documentation may be rendered in any number of forms. Some common forms include:

- Requirements document, which may be a business requirements document and/or a functional requirements specification and/or a system or software requirements specification, etc.;
- Deck of user stories written;
- Set of use cases with accompanying nonfunctional requirements; or
- List of items on a product backlog.

The format of the solution documentation is defined in business analysis planning.

4.11.3.1 Requirements

Product requirements are written at different levels of detail and are associated with different requirement types, for example, business, stakeholder, solution, and transition requirements, where solution requirements are further categorized as functional and nonfunctional. Within any of these types, requirements can be written into progressive levels of detail, and when this occurs, these can be documented in a hierarchy or with a numbering convention that demonstrates the hierarchical progression.

While product requirements describe what is being built or the outcome of the project or solution to the business problem, project requirements describe the constraints and necessities for successful completion of the project.

> *Example*—For example, product requirements describe the length and width of the sidewalk to be constructed in front of a building, along with such aspects as color and texture. The project requirements for laying the sidewalk could include the number of laborers required, qualifications of the laborers to handle the equipment, size of the equipment, time frame for usage, and any restrictions on labor hours.

Collaboration Point—Product requirements are the responsibility of the business analyst. Project requirements are the responsibility of the project manager.

4.11.3.2 Categorization

Requirement categories are used to help group and structure requirements within the documentation. The requirement categories may be determined before starting the documentation effort or the requirements may be documented first and then the categories decided after. When choosing the categories later, they can be determined based on the actual information gathered. Selecting the categories earlier may provide a structure for use in organizing how to elicit information from stakeholders.

The process of categorization helps expose vague, misstated, ambiguous, or otherwise poorly written requirements. When the business analyst is unable to place information or a requirement in a category, the requirement is likely to be invalid and may need to be revised, expanded, or removed. Categorization used in this way filters out the bad or poorly written requirements.

Examples of possible filters are:

- **Scope filter.** Determine whether a requirement or information is in scope, out of scope, or unknown. Unknown refers to requirements that are possibly invalid and need to be revised or omitted.
- **Functional filter.** Once the functional categories have been determined, any defined functional requirement not fitting into one of the categories can be filtered out, revised, or discarded.
- **Prioritization filter.** An arbitrary level of priority (e.g., needs, wants, and desires), is defined and used as a filter to determine which requirements stay or are removed.
- **Testability filter.** All requirements need to be testable, and all requirements should be examined to determine if they are testable. Any requirements that are not testable are filtered out and need to be revised.

  Note: Filters may indicate the need for an additional classification, implying that the original list of filter elements is incomplete.

4.11.4 Requirements Specification

The requirements specification is a common form for documenting requirements. The requirements specification attempts to specify all circumstances, conditions, actions, reactions, results, and error conditions that could possibly occur in the defined solution. The concept is to create a manual that is followed by the development team to create the solution. Many books have been written on the subject of how to construct a requirements specification, and how the individual requirements within that specification should be written. There are several available standards that can be followed. Each attempts to circumscribe the contents and form of the requirements specification.

Requirements specification is a generic term that includes all documents that contain requirements. These requirements may be high-level, business-oriented wants and needs, or very detailed specifications required to build the new product or service.

Formal requirements specifications are more lightweight on projects using agile, lean, or hybrid methods. These teams may produce a one- to two-page specification and may include user acceptance testing and trace matrices to holistically describe the product.

4.11.4.1 Documenting Assumptions

An assumption is a factor that is considered to be true, real, or certain, without proof or demonstration. Analysis starts with assumptions and these assumptions are either proven or disproven over the course of a project. There are many situations where assumptions are valid and warranted such as:

- Complete information is unavailable; for example, the business analyst assumes that the technical resources to implement the solution will be available when needed.
- The success of the project is dependent upon something happening in the future; for example, an increase in customer population, or a change in consumer interest.
- Assumptions are made based on factors that exist currently, but those factors may not exist in the future. The common assumption about a project, for example, is that all members of the team will stay on the project until the project is completed. The longer the project, the less likely that this assumption is true.

While assumptions may not hold true over the course of the entire project, they impose a level of risk. The business analyst identifies a contingency for each documented assumption so there is a course of action should that assumption turn out to be false. Assumptions, once documented, should be monitored by the business analyst and project manager. As the project progresses and more information is known, assumptions may be deemed invalid or not relevant. When this occurs, the assumption is closed and a reason is provided as to why the assumption is no longer valid.

4.11.4.2 Documenting Constraints

The project management view of a constraint is that it is a limiting factor that affects the execution of a project, program, portfolio, or process. In business analysis, a constraint is a limiting factor placed on the product or solution. Therefore, there are two levels of constraints: product or solution constraints and project constraints.

> ***Collaboration Point***—The business analyst is primarily concerned with the product or solution constraints, although frequently project constraints are voiced by stakeholders during elicitation. Therefore project constraints are documented by the business analyst and passed along to the project manager for follow-up.

While some consider all requirements to be constraints on the solution because both constraints and requirements need to be met, solution constraints are often treated as a specific category of requirement. Constraints are considered by some to be a form of nonfunctional requirements, while others prefer to think of nonfunctional requirements as the origin of some design or implementation constraints.

The difference between constraints and requirements is that requirements are written in the positive voice, because they define something that should be done. Requirements should not state that something not be done. While requirements define the solution by stating the "what," solution constraints are typically written in negative

or constraining language and limit the solution by stating actions that cannot be performed. These constraints are placing limitations on how the problem described by the requirements is solved. Solution constraints are best highlighted by being placed in a separate and distinct section of the requirements document.

Solution constraints may fall into a few specific categories:

- Geography,
- Regulations,
- Organizational policy, and
- Culture.

Some examples of solution constraints are:

- Restricting access to sales information to the region that the customer is located in (geography),
- Prohibition against using certain building materials (regulation),
- Security restrictions preventing the use of certain data or systems by unauthorized personnel (policy), or
- Limitations on the amount of change that the organization can withstand over any period of time (culture).

Project constraints may include:

- Direction to use predetermined equipment, organizational standards, or a preferred supplier;
- Restrictions, such as resource utilization, message size and timing, software size, maximum number of and size of files, records, and data elements; or
- Time (deadline), cost (budget), and scope.

4.11.5 Guidelines for Writing Requirements

Information needs to be transcribed into high-quality, well-formatted requirements. Requirements that are well written are of higher value to the solution developer and overall project team, because these will be clear, concise, and reduce conflict and confusion on what needs to be delivered.

How requirements are written and documented and the guidelines that help structure them are dependent on the selected project life cycle. Requirements are often written in a text-based format when developing business requirement documentation or solution requirement documents, and written in the format of user stories for projects that follow an adaptive life cycle. The guidelines for both formats are presented here. Requirements can be more than text and may also take a visual form as well.

4.11.5.1 Functional Requirements

A well-formatted requirement consists of the following elements:

- Condition,
- Subject,

- Imperative,
- Active verb,
- Object,
- Business rule (optional), and
- Outcome (optional).

Example—A well-formed detail level requirement might be as follows: When the new account button is pressed (condition), the system (subject) will (imperative) display (active verb) the new account entry screen (object) allowing the creation of a new account (outcome).

The following characteristics serve as a checklist when reviewing requirements to ensure they are of high quality. The guidelines address the writing and not the format of the requirement and are applicable to any format or rendering of the solution document. The following characteristics are present in all requirements when they are of a high level of quality:

- **Unambiguous.** Clarity is key; therefore, the business analyst should take steps to ensure that the written requirements are not ambiguous. When two individuals disagree on the meaning of a requirement or when an individual interprets a requirement differently from its intended meaning, then the requirement is ambiguous. The requirement needs to be rewritten to remove the ambiguity. If the requirement is not clear before requirements are baselined, stakeholders may have different interpretations of the requirements. Ambiguity may result in the solution team building the wrong solution component, and others who base their work from the requirement as stated may perform their work incorrectly. A written requirement should be reviewed to see if it can be stated in a simpler or more straightforward manner (see Table 4-8).
- **Precise.** As a whole, the solution document states precisely what the solution to the business problem is—no more, no less. Precision also refers to choosing the right words. With adaptive life cycle methods, the business analyst should specify at what point the requirement will need to be precise, as the requirements will unfold into details through incremental elaboration; therefore, it is understood early on that all details may not be known. See Table 4-9 for examples of precise and imprecise language.

Table 4-8. Example of Ambiguous vs. Unambiguous Requirements

Note: This requirement construct is in a format that would be used for text-based requirements written for a project following a predictive life cycle. Adaptive life cycle projects may use user stories and use cases and use a different format.

Ambiguous Requirements	Unambiguous Requirements
The system shall check the name field to be only alphabetic and the address field to be either alphabetic or numeric but containing only addresses in the U.S. or Canada, and the quantity field to be only numeric	3.4.1 The system shall validate that 3.4.1.1 The name field is alphabetic 3.4.1.2 The address field is either alphabetic or numeric 3.4.1.3 All addresses are in the U.S. or Canada only 3.4.1.4 The quantity field is numeric only
The system provides identification of the employee when passing through the reader	3.9.12 When the employee passes through the reader, the system displays the photograph of the employee on the monitor

Table 4-9. Examples of Precise and Imprecise Language

Note: In this example, precision is obtained by specifying a business rule that is best maintained in a business repository and not hard-coded within the software.

Imprecise	Precise
9.2.1 When the department code entered does not match the department code on file, the system will display an error message.	9.2.1 When the department code entered does not match the department code on file, the system will display "invalid department code."

Table 4-10. Examples of Inconsistent and Consistent Language

Inconsistent	Consistent
17.1.4 The security system will	17.1.4 The security system will
22.4.9 The new security system will	22.4.9 The security system will
33.9.11 The secure card system will	33.9.11 The security system will
34.12.12 The R/F security system will	34.12.12 The security system will

- **Consistent.** Each requirement should be included one time in the solution document to avoid contradiction and redundancy. The requirements should not be in conflict with other requirements within the documentation set. The language needs to be consistent throughout. The business analyst maintains consistency through rewrites, revisions, changes, and modifications to the solution document. These revisions occur naturally in the iterative nature of business analysis. As more information is uncovered during elicitation, the business analyst analyzes it and incorporates it into the documentation. A traceability matrix helps to ensure consistency. Traceability is used to verify consistency and is achieved when determining the relationships between requirements. Traceability and the traceability matrix are discussed in detail in Section 5 on Traceability and Monitoring.

  Conflicting requirements are not unusual when there are multiple business analysts working on the same set of requirements and performing elicitation separately. One way to prevent contradicting requirements is to assign only one business analyst the responsibility for writing the finished document. Inconsistencies are also introduced in requirements when multiple terms are used to mean the same thing. For example, when referring to the result of the project as the "new system," the new "accounts payable system," and the "financial system" within the same set of documentation; those reading will assume that the document refers to three different systems. Even though it may be repetitive to use the same terminology, it ensures consistent and unambiguous requirements. See Table 4-10 for examples addressing consistent language.
  Contradictions occur when stakeholders have opposing requirements. For example, two different business units or constituencies may each want their own requirements regardless of how their requirement may impact another stakeholder group. There are a few ways of resolving this particular situation:

  - The most direct route is to bring the contradicting parties together in the same room and have them work through the inconsistency.
  - Information can be added to one requirement, which details special circumstances that remove the contradiction.

Table 4-11. Examples of Correct and Incorrect Inclusion of Requirements

Incorrect	Correct
7.0 Security	7.0 Security
7.7.1 The password will be at least 8 characters in length	7.7.1 The password will be 8 characters in length
11.0 User interface	11.0 User interface
11.9.13 The password will be no less than 8 characters in length	11.9.13 The password length is defined in section 7.7.1

- It may be necessary to support both requirements; in these cases, requirements for each user group are captured in the form of stakeholder requirements.

Example—For example, when one business unit requires 4000 transactions per day and another business unit requires 2000 transactions per day, the requirement will be consistent if stated that 4000 transactions a day are required during the holiday season from October to December 31, and 2000 transactions a day are required for the remainder of the year. The requirement may be rewritten to add more clarity such as: 4000 transactions a day are desired, but 2000 transactions a day are mandatory.

Within a single software requirements specification, it is possible to introduce ambiguity by repeating information. When requirements are repeated, there is a risk that a change is reflected in one requirement and left unchanged in the duplicated one. One way to avoid ambiguity incurred through redundancy is to remove the redundancy (see Table 4-11 for an example).

- **Correct.** Each requirement should accurately describe the functionality to be built. Correctness is not absolute; the solution document is only as correct as the information that has been obtained up to that point. As more information is uncovered, adding new information makes the solution document more correct by removing some assumptions, clarifying ambiguities, and adding new information in a progressive, elaborative approach.

  The following basic rules help to ensure correct requirements:

  - Only the product stakeholder can confirm that a requirement is correct. Correctness is established through frequent review and confirmation sessions with the sources of information. Correctness is the purview of the business community.
  - In general, no single requirement should be committed to the solution document until it has been confirmed by a second source. In the case of requirements documentation, the second source may be another individual from the business community, or a different way of gathering information (e.g., gathering information about a process through observation and then interviewing the process workers who have been observed for confirmation on what has been seen).

- **Complete.** Completeness is also not absolute. The requirements can be made more complete with more information. Therefore, the guidelines that apply to correctness also apply to completeness. The business analyst should ensure that enough information is gathered and documented to complete the requirement; however, too much information makes the requirement difficult to follow and convey.

  Guidelines concerning requirements completeness are as follows:

Table 4-12. Examples of Complete and Incomplete Requirements

Incomplete	Complete
The card reader shall be of the same dimensions as indicated by the card size and consistent with industry standards	54.1 Card reader dimensions TBD by April 3 (by John Doe, Security Architect)
Terminate a session after the number of incorrect passwords exceeds the maximum allowed	24.2.2 Terminate a session after the 3 incorrect passwords have been entered

Table 4-13. Examples of Measurable and Not Measurable Requirements

Not Measurable	Measurable
There will be no more than 6 training classes per employee	Each employee will have not less than 2 and no more than 6 training classes residing on their professional development profile
The new production line shall be efficient	The new production line shall produce an average of 5000 bottle caps per day

- Document all known requirements, especially those that are confirmed by the stakeholders. This includes all conditions that apply to a requirement.
- Include in each requirement all of the information necessary for the solution team to design, build, and test the solution component. A requirement is said to be "self-contained" when this is true.

A requirements specification is complete when it contains the following (see also Table 4-12):

- All necessary requirements,
- Responses specified for all inputs,
- Requirements that produce all necessary outputs, and
- Labels and references to all figures, tables, and diagrams.

Note: What constitutes completeness is dependent on the selected project life cycle.

Because of necessary assumptions, requirements may exist in an incomplete form. Use of the term "to be determined (TBD)" is acceptable for use provided there is a date when the information is to be determined and, optionally, the name of the person who is responsible for determining it. These should be resolved for a given portion of the requirements before proceeding with construction.

- **Measurable.** Each requirement needs to be independently measurable. A requirement that is measurable provides the necessary detail to understand the criteria for testing. Measurability is usually a prerequisite to testability; a requirement that is not measurable cannot be tested (see Table 4-13 for examples).
- **Feasible.** Feasibility was discussed in detail in Section 2 on Needs Assessment. The focus of the feasibility analysis performed at the forefront of the business analysis work pertains to the work to determine the feasibility of various solution options. The feasibility discussed in requirements elicitation and analysis pertains to the feasibility of each requirement. Feasibility here is conducted at a much more specific and detailed level.

  The same categories of feasibility used in the needs assessment when evaluating a solution option can be applied here to evaluate the feasibility of a requirement. For example:

- *Operational feasibility.* When the solution requirement is met, will the implementation of it within the solution be supported by all stakeholders who use the new product or solution? Ensure a solution requirement for one stakeholder group does not make the use of the product inefficient or unusable to another stakeholder group.
- *Technology/system feasibility.* Can the requirement be fulfilled based on the technologies that have been selected for the solution? While the solution as a whole was assessed for technical feasibility, the focus here is on each specific requirement. Involve the solution development team to ensure that each requirement is technically feasible. Technical feasibility is more easily kept in check with adaptive project life cycles because the project team is working on a small amount of the solution at a time and is collaborating daily, whereas with the predictive life cycles, there is a risk that the business analyst is not interacting with the solution development team often enough. When following a predictive or iterative life cycle, ensure requirements are evaluated for feasibility by the solution development team before they are baselined. Business stakeholders will be more disappointed to be informed late in the process that a highly desired requirement was not technically feasible than learning about it early in the process when the requirement was provided.
- *Cost-effectiveness feasibility.* Does the cost to fulfill the requirement make sense with respect to the value the requirement will deliver to the business? Cost may be one of the requirement attributes that is being tracked. Work with the solution development team, business stakeholders, and the project manager to ensure the cost-effectiveness of each requirement is analyzed. A requirement makes sense when the value it provides is greater than the cost to implement it.
- *Time feasibility.* Can the defined requirement be met within the time allocated for the project phase or should the feature be considered for a future release? A project delay may occur from a single requirement; therefore, it is better to know up-front when a requirement will require a level of effort by the solution development team that exceeds the time allocation for the entire development phase.

There is no one single factor such as time or cost that determines the feasibility of a solution option or evaluates the feasibility of a requirement. Feasibility is best analyzed according to a variety of factors.

- **Traceable.** Traceable requirements are those that can be mapped back to the source of the requirement and mapped forward through the development life cycle to a test case that proves that the requirement was successfully satisfied. Requirements may also be traced between requirements and back to business goals, objectives, and higher level requirements.

  It is important to be able to trace any given requirement back to a source in the event that there are changes to the requirement or other changes that impact the requirement. The business analyst uses the source to identify who to contact when the requirement changes.

  Within predictive life cycle projects, from a project management perspective, traceability provides a fairly accurate estimate of the level of completion for development. When all of the requirements can be traced to test cases through design and build, the percent complete, and more importantly what remains to be

completed can be determined by the number of test cases that have been successfully executed. When 100% of the test cases have been executed, then 100% of the requirements have been satisfied, at least in the build phase. Traceability is further discussed in Section 5 on Traceability and Monitoring.

- **Testable.** Requirements should be written in a way that allows them to be tested. When a requirement is not testable, it is typically because the requirement is vague, unclear, ambiguous, or has violated some other principles or writing guidelines for quality requirements. Testable requirements allow for an assessment of pass/fail. If the requirement is written to be measurable, then the requirement can be tested.

  Confirming the testability of a requirement does not mean creating or writing the test case for execution during the test stage. Confirm only that a test can be created to verify that the requirement has been satisfied. Sometimes the evaluation criteria are constructed in the case of the development of user stories. For more information on the evaluation activities performed for solution validation, see Section 6 on Solution Evaluation.

4.11.6 Prioritizing Requirements

How requirements are prioritized should be fully defined in the business analysis plan. Section 3 on Business Analysis Planning discusses the criteria that the project team may use to prioritize the product requirements. The business analysis work in requirements elicitation and analysis is performed to use one or more prioritization techniques in order to facilitate priority decisions from the key stakeholders. The key stakeholders here are those stakeholders who have the authority to prioritize requirements as specified during planning.

4.11.6.1 Prioritization Schemes

There are several methods to evoke prioritization of requirements from the stakeholder community. Many business stakeholders may find it difficult to make decisions regarding prioritization, as they may see all of the requirements as equal, or at least the requirements they provided. Drive the prioritization decisions from the business by using one or more of the following techniques to help support the prioritization activity:

- **MoSCoW.** MoSCow establishes a set of prioritization rules which are:
  - Must haves (fundamental to project success),
  - Should haves (important, but the project success does not rely on them),
  - Could haves (can easily be left out without impacting the project), and
  - Won't haves (not delivered this time around).
- **Multivoting.** There are many forms of multivoting that are designed to gain active participation from stakeholders. This technique provides a process for participants to apply votes to a list of items to determine an answer based on number of votes received. When using multivoting for setting priorities, the requirement with the most votes is deemed the higher priority item.

Example—When applying multivoting to determine a list of prioritized requirements, each stakeholder could be given a set of sticky dots in three different colors: red for high priority, blue for middle priority, and yellow for low priority. With the requirements on a whiteboard or flipchart, the stakeholders place one or more of their sticky dots next to the requirement they believe merits the priority. A participant can place all of the red sticky dots next to a requirement believed to be of the highest priority. Once all the dots have been placed, the dots are totaled and the requirements prioritized. The game-like atmosphere of this approach makes it easy and fun for stakeholders to prioritize requirements.

- **Time-boxing.** Time-boxing is a prioritization technique that is used when the project has a fixed timeline and the timeline is not negotiable. Time-boxing approaches the prioritization of requirements by analyzing the amount of work the project team is capable of delivering during a prescribed period of time. The project team determines the scope based on what work can be completed within the fixed window of time. If the time-box is 90 days, the project team evaluates the list of requirements and determines what can be delivered within that 90-day window. Time-boxing is often used with other prioritization techniques, such as MoSCoW, to ensure that the requirements time-boxed into the product release are those the business has selected as the highest priority or highest valued. A variation of this technique uses money instead of time to determine which requirements can be delivered based on a budget.
- **Weighted ranking.** Weighted ranking begins in business analysis planning before the possible solution options are listed. Prior to performing weighted ranking, the business analyst facilitates a decision regarding which criteria the decisions will be based on. Possible options are efficiency, ease-of-use, attractiveness, etc. Once a short list of criteria has been selected, the criteria are ranked with a score. The highest score is assigned to the criterion considered to be most important, thereby creating the weighted rankings. Weighted ranking is further defined in Section 2 on Needs Assessment where it is used to rank solution options.

4.11.7 Technical Requirements Specification

Projects may require more detailed technical documents (e.g., software IT, construction, manufacturing projects to name a few). The requirements are documented in the format of a technical requirement specification.

Technical specifications may contain such elements as:

- Wireframes describing the appearance of a user interface,
- Screen mockups specifying the details of a user interface screen,
- Data models and schema,
- Detail process flows such as data flow diagrams or activity diagrams, and
- Detailed requirements that include technical references.

Collaboration Point—For IT projects, some organizations have a systems analyst who produces the technical specification, dividing the requirements documentation between the business analyst and the IT analyst. The business analyst then produces the business requirements specification and the systems

analyst prepares the technical requirements specification. In this practice guide, the role of business analyst is referred to in the general sense; therefore, the work to produce the technical requirements specification is included as a possible documentation activity, recognizing that the documents that are produced will be dependent on the selected project life cycle.

4.11.8 Documenting with Use Cases

Use cases may be used by an organization in addition to a functional requirements specification or used instead of producing a separate functional requirements specification. A use case may represent one or more functional requirements. Use cases may supplement text-based requirements and are developed for areas of the system where the interaction of the system and the user are more complex. Use cases may be used when there are multiple paths and scenarios that the system needs to accommodate. Instead of creating a functional requirements specification, some organizations may decide to use cases and will choose to use models, such as those discussed in this practice guide, instead of text-based requirements. The decisions regarding the preferred documentation approach for the project is made in business analysis planning. Use cases are explained in Sections 4.10.7.5 and 4.10.8.2.

4.11.9 Documenting with User Stories

A documented user story is sometimes written on an index card. Writing the story on a card enforces brevity and concision. When cards are not used, maintain the stories in a document, spreadsheet, or requirements management tool. When packaged together, user stories represent a high-level version of solution requirements. User stories are a documentation method for breaking down features into manageable parts and provide a simple and effective mechanism to segment a complex set of features into simple, definable elements. User stories are explained in Section 4.10.8.3.

4.11.10 Backlog Items

A backlog is a prioritized listing of product requirements and deliverables to be completed, often written stories, and prioritized by the business to manage and organize the project's work. As the use of adaptive project life cycles has become more prevalent, the concept of building a backlog has gained in popularity and across approaches. Where backlogs are commonly leveraged to contain only user stories, the term can be used more broadly as backlogs may contain use cases, requirements, and defects to be fixed, in addition to the user stories. Regardless of the content stored in the product backlog, to ensure clarity, each item should be written with the same care and follow the same guidelines as a requirement expressed in a business requirements document.

In agile approaches, the business analyst is often assigned to help the product owner groom the product backlog, which involves adding and removing backlog items and reprioritizing based on changing business conditions and priorities.

4.12 Validate Requirements

Validation is defined as the assurance that a product, service, or system meets the needs of the customer and other identified stakeholders. In business analysis, requirements validation is the process of ensuring all

requirements accurately reflect the intent of the stakeholder; thereby ensuring the requirements meet their expectations.

4.12.1 The Concept of Continual Confirmation

Confirmation of requirements can occur whether the requirements are written down in a requirements document or are displayed on a whiteboard in a requirements workshop. Confirmation is not an activity that is performed once at the end of the requirements process. Confirmation occurs whenever the business analyst reviews the information gathered during an elicitation session with the stakeholder or any part of the developing solution is shown to the stakeholder. Confirmation is not approval—only authorized stakeholders can approve. Confirmation means obtaining agreement from the stakeholders that the solution under development is good and will accomplish the objectives.

It is helpful to demonstrate parts of the solution as it is being completed. It is easier to obtain low-level confirmation on a continuing basis than to try to have the entire solution document confirmed and approved at the same time. When the requirements are broken up into small, self-contained units, the review sessions involve fewer participants and take less time.

It may also be helpful to divide the requirements document up by functional area and ask stakeholders to confirm the requirements that impact their area. This requires more work, but stakeholders will appreciate being able to focus specifically on their own requirements. There is a risk, however, that the separation of requirements may not be correct, and stakeholders impacted by a group of requirements may not be provided with all of the requirements that will impact them.

4.12.2 Requirements Walkthrough

Requirements walkthroughs are used to review the requirements with the stakeholders and to receive confirmation that the requirements as stated are valid. Valid requirements accurately reflect what the stakeholders are asking the solution developers to develop.

To conduct a requirements walkthrough, the business analyst schedules the session providing the stakeholders enough time to prepare. Reviewers may be asked to read the materials before attending and, by doing so, the reviewers will be able to:

- Think about the stated solution ahead of time, better preparing their feedback,
- Avoid providing emotional reactions to the material in the session, and
- Take time to discuss the materials with their organizational units and peers so that feedback provided is representative of the larger stakeholder group.

The flow of information in a review session should come from the reviewers as much as possible. This session is one of the final opportunities for the stakeholders to raise questions, seek clarity, and voice concerns. The business analyst can use this session as an opportunity to field questions and seek to close down any open requirements-related issues in preparation for receiving final approval of the requirements.

4.13 Verify Requirements

Verification is the process of reviewing the requirements for errors and quality. Verification may occur before or after validation. Verifying requirements after they have been validated or confirmed by product stakeholders ensures that only the requirements that are considered to be "good" requirements are verified. Validation is concerned about ensuring that the requirements solve the problem—verification is not.

Verification is performed by members of the solution team to ensure that the requirements meet quality standards or any other business analysis deliverable in the process meets the standards of excellence for the organization. There are two types of verification processes: peer reviews and inspections.

4.13.1 Peer Review

Peer reviews can be conducted at any point during the requirements process. In business analysis, a peer review is a formal or informal review of the requirements by the peers of the business analyst. Business analysts may include other practitioners, such as their manager or someone in the Business Analysis Center of Excellence. Quality assurance or testing resources also make great reviewers.

The purpose for the peer review is to ensure that the written business analysis deliverables, including all forms of requirement documentation, are in compliance with the organizational standards and generally held principles of requirements writing. Section 4.11.4 contains examples of these guidelines.

An informal peer review may be held with a couple of business analysts before conducting a requirements review meeting with business stakeholders. This helps to ensure there are no glaring errors in the requirements that could cause problems during the review. Peer reviews may also be held after the review sessions with the business stakeholders to ensure the final document is understood and accepted by the solution team and development community in general.

Conducting a peer review is fairly straightforward. All reviewers are provided with a copy of the document and asked to review the document prior to attending the peer review session. The requirements are reviewed as a group with each reviewer pointing out issues or problems in the document. A form of screen-sharing technology can be used to display the document and make changes to it during the meeting as issues are discussed, thereby saving time and ensuring the errors raised are properly fixed. When a formal session is not conducted, the document can be distributed to the peer review team electronically, asking the reviewers to track their changes in the original document. For organizations who use a requirements management tool, there are a number of features that can be used to track the comments provided by document reviewers.

> ***Collaboration Point***—Ask testers and those involved in developing training manuals to be part of the verification review. This provides these team members with the project background and context early on. These team members will focus on the details and check for consistency, completeness, and testability of the requirements, which is beneficial for the verification review.

4.13.2 Inspection

Inspections are a more rigorous form of a peer review. Inspections were initially targeted for hardware development and were later modified to be applicable for anything that needs to be reviewed for accuracy, completeness, and relevance, etc.

In a predictive life cycle process, requirements inspections are typically performed once at the end of the requirements process when the document is ready for final approval. Often, inspections are performed after all of the business stakeholders have had a chance to review and confirm their respective portions of the requirements. The objective is for the business to ensure that the requirements are acceptable in their current form. Inspections can also be performed on groupings or chunks of completed requirements.

The inspection is performed by peers who participated in the creation and documentation of the requirements and are the recipients of the requirements document. Business stakeholders and management are excluded from an inspection session, especially the line management of the inspectors.

A requirements inspection checklist could include the following items:

- Are all internal cross-references to other requirements correct?
- Are all requirements written at a consistent and appropriate level of detail?
- Do the requirements provide an adequate basis for design?
- Is the implementation priority for each requirement included?
- Are all external hardware, software, and communication interfaces defined?
- Have algorithms intrinsic to the functional requirements been defined?
- Is the expected behavior documented for all anticipated error conditions?

A checklist differentiates an inspection from a peer review. A peer review depends on the knowledge of the reviewers whereas the inspection uses a checklist of known defects for the product being inspected. Inspections also follow a more rigorous process than peer reviews and use a series of rules to govern how they are performed.

4.14 Approval Sessions

Approval sessions are conducted separately from confirmation sessions. After the solution is confirmed and validated, the business analyst obtains signoff on the requirements. Signoff may be formal or not and may be predetermined in business analysis planning.

When signatures are sought, there are three signatures that are usually requested:

- *Business owner,* such as the executive in charge of the business area, who agrees that the requirements represent a complete and accurate solution to the problem.

- *Solution team recipient,* who accepts the document and acknowledges that it is sufficient to build the documented solution.
- *Business analyst,* who prepared the deliverable.

Obtaining approval for the requirements should be a fairly automatic procedure. It is rare for a person of authority, such as a vice president or director of the company to take the time to thoroughly read and analyze a complete set of requirements for any given project. They may ask someone closer to the project whether the requirements are acceptable. Since the requirements are confirmed to be correct, accurate, understandable, and implementable, etc., by all of the individual constituencies, obtaining approval from a senior manager should be a routine process. Problems in obtaining approval arise when the lower-level managers or process workers have not seen the requirements yet and are unable to provide a positive opinion about them to the senior manager. For more information about the requirements approval process, see Section 5.4.

4.15 Resolve Requirements-Related Conflicts

Conflicts may arise at any point in the business analysis process. Whether the conflict is between business units voicing opposing views of what the solution should be or a solution team and a product stakeholder disagreeing on the way to solve the business problem, the first order of business is to determine what the problem is that the parties are attempting to solve.

The business analyst mediates the situation by discussing the differences and by understanding the points of view of each stakeholder. Several discussions may need to occur before a resolution is reached. When unable to reach a decision, the issue needs to be escalated. The process for making decisions, resolving requirements-related conflicts, and the escalation path to follow when negotiating efforts fail should have been defined during business analysis planning.

Business analysts require soft skills in negotiation and need to learn how to bring opposing sides to consensus. Facilitating a win-win solution is key. The business analyst also should be capable of analyzing the reasons as to why the conflict exists. There are numerous techniques that can be used to help a team reach a decision or resolve a conflict. Techniques remove subjectivity and emotion from the process. A few examples are listed in Sections 4.15.1 through 4.15.3.

4.15.1 Delphi

The Delphi technique is an information-gathering technique used as a way to reach consensus from experts on a subject. Experts on the subject participate in this technique anonymously. A facilitator uses a questionnaire to elicit ideas about the important points related to the subject. The responses are summarized and are then recirculated to the experts for further comments. Consensus may be reached in a few rounds of this process. The Delphi technique helps to reduce bias in the data and prevents any one person from having undue influence on the outcome.

The Delphi method relies on peer pressure and the wisdom of crowds to produce the correct decision or solution. This technique works well for teams operating from diverse locations.

4.15.2 Multivoting

When applying the multivoting technique to resolve a conflict, the team brainstorms to generate a possible list of options for resolving the conflict. The team decides how many items will be on the final list. All of the items remaining after a first cut of the brainstormed answers or solutions are then numbered. Each participant receives a limited number of votes and is asked to place the votes against the unranked choices. For example, when provided with five votes, a participant is able to place three votes on one choice and two on another. When all members of the team have voted, the votes are tallied and the results posted. When there is a clear decision, the process ends. Otherwise, the total list is adjusted by eliminating those at the bottom and the process is repeated, this time with fewer votes per person. In the end, the decision is based on the option with the highest score.

4.15.3 Weighted Ranking

The same weighted ranking technique used to rank solution options in needs assessment and to prioritize requirements in requirements elicitation and analysis, is applied to resolve requirements-related conflicts. The technique is applied in the same manner: options are listed, ranked, and voted upon. The option with the highest score is selected. Options are typically solution-related. For example, in the case of two requirements that completely conflict with each other, this technique does not compare the requirements to each other, but instead compares the differences between the solution options that represent the requirements.

5

TRACEABILITY AND MONITORING

5.1 Overview of this Section

From the perspective of the *PMBOK® Guide* – Fifth Edition, traceability and monitoring consist of the activities completed to ensure that requirements are approved and managed throughout the project life cycle. During traceability and monitoring, the traceability matrix and associated attributes are created and applied to help monitor and control the product scope. Approved requirements are baselined and tracked. As new requirements surface, these are documented, added to the traceability matrix, assessed for their impacts to the project and product, and presented to stakeholders for approval. Throughout traceability and monitoring, the status of all requirements is communicated using the communication methods defined and approved within the business analysis plan.

The kind of thinking that is inherent in traceability and monitoring applies to all projects and all life cycles. Thinking about the relationships between requirements and their relationships to other project considerations, such as tests and releases, is critical for ensuring project consistency and completeness. Traceability principles that enable change impact analysis are the basis for confirming fulfillment of objectives and ensuring test coverage. Traceability enables the discovery of missing and extraneous requirements. There is a need to track and monitor completed requirements, no matter what type of life cycle is used for a project or what type of format is used to document the requirements. Traceability should be maintained, at a minimum, for the entire duration of the project, even when someone with a business analysis role is no longer associated with the project.

Formal and comprehensive traceability and monitoring requires up-front effort for setup but only provides benefits when there is an ongoing commitment of effort to maintain it and when it is used and referred to by the stakeholders. Organizations that have identified a need for formal traceability and are willing and able to invest in it are more than paid back for that investment because traceability makes it easier to manage requirements. That said, not every organization or project may need formal and comprehensive traceability. Moreover, some degree of traceability can be achieved in less formal ways and may be sufficient for the work at hand.

While some examples of streamlined approaches to traceability are mentioned, this section primarily provides information that would enable an organization to undertake formal, comprehensive traceability and monitoring in a way that is in alignment with *PMBOK® Guide* – Fifth Edition principles. It highlights the value that these techniques and the underlying thought processes bring and the risks associated with streamlining, as well as the risks of overuse. Practitioners and organizations can use it as a starting point to consider their own traceability needs and risk acceptance and to determine the optimal amount of traceability and monitoring appropriate for their work.

5.2 Traceability

5.2.1 What is Traceability?

Traceability provides the ability to track product requirements from their origin to the deliverables that satisfy them. Traceability is sometimes qualified as bidirectional or forward and backward, because requirements are traced in more than one direction. Not all projects require the same amount of traceability; therefore, the specific deliverables that will be traced for the project are determined during business analysis planning. Generally, more complex projects require more traceability. A project in a heavily regulated industry or one with numerous components, interfaces, risks, and stakeholders will likely require more traceability than a project without these characteristics.

From the perspective of the *PMBOK® Guide* – Fifth Edition, traceability includes, but is not limited to tracking the following requirements:

- Business needs (business problems or opportunities), goals, and objectives;
- Project objectives;
- Project scope/WBS deliverables;
- Product design components;
- Product development components;
- Test strategy and test scenarios;
- High-level to more detailed-level requirements;
- Detailed to higher-level requirements; and
- Different types of functional requirements that are related to each other. For example, adding a new customer type requires changes to several business processes that use customer data.

Additionally some practitioners trace:

- Use cases to acceptance tests, as well as to other types of requirements, and
- Models and diagrams to their related requirements.

For projects using an adaptive life cycle:

- Epics or user stories may be traced to features and acceptance tests.
- A Kanban Board may provide some amount of traceability.

Collaboration Point—Decisions about how the traceability process will be conducted are determined during business analysis planning. Conducting minimal traceability can incur project risks, and using an extensive traceability process can consume a lot of time and require extensive management of the traceability links. The business analyst should work with the project manager to determine how much traceability is appropriate for the project, because traceability impacts the usage of resources and the level of project risk.

5.2.2 Benefits of Tracing Requirements

Tracing requirements provides significant benefits:

- **Helps to ensure that each requirement adds business value.** Tracing each requirement to the business need, goals, and objectives helps ensure its relevancy. When the requirement helps to solve the business problem, takes advantage of a business opportunity, or meets at least one goal or objective, it is relevant and is an indication that the end product will add value. On the other hand, when the requirement is unable to be traced to a business need, goal, or objective, its relevance needs to be analyzed. In this case, the business analyst should consider such things as:
  - Have the business needs, goals, and objectives been articulated in enough detail? A requirement may be relevant, but there may be an incorrectly stated problem or missing objective.
  - Are there missing requirements? Project objectives with no associated requirements indicate that the objective will not be met. Without defined requirements, the end product, service, or result will not add the value that the organization anticipates.
  - Does a specific project objective belong in the project? When there are project objectives without requirements, the objective is misaligned with the current project, in which case it belongs elsewhere or should be eliminated.
- **Helps to meet customer expectations.** Traceability provides a means to track requirements throughout the project life cycle, helping to ensure that approved requirements are delivered at the end of the project. For example, a data field on a user interface (UI) is an approved requirement, but if it is not developed, tested, or delivered, the expected data field will be missing from the UI in the implemented solution.
- **Helps to manage scope.** It is more difficult for requirements that do not add value to make their way into the product, because approval is provided only to requirements that add value.

  Predictive projects provide a better case for formal traceability than adaptive projects; however, the life cycle model is not the only factor that determines the optimal amount of traceability. Other factors include whether or not the business is highly regulated, organizational policies and processes, and the degree to which traceability is actually used.

  Collaboration Point—There is often confusion between project managers and business analysts as to who manages scope. The project manager is responsible for managing project scope while the business analyst is responsible for managing product scope. The process of tracing requirements is a clear example of the involvement that the business analyst has in scope management at the product level. Several models that are presented in Section 4 on Requirements Elicitation and Analysis also demonstrate how the business analyst analyzes product scope while modeling.

5.2.3 The Traceability Matrix

Organizations often trace their requirements using a structure called a traceability matrix. A traceability matrix is a grid that allows for the linkage of product requirements from the source to the deliverables that satisfy them

throughout the project life cycle. The implementation of a requirements traceability matrix supports the goal that each requirement adds business value by linking it to the business and project objectives. It provides a means to track requirements throughout the project life cycle, helping to ensure that requirements approved in the requirements documentation are delivered at the end of the project. The matrix also provides a structure for managing changes, thereby helping to manage the product scope.

For formal traceability, a traceability matrix contains a short description of each requirement and facts about the requirements, which are called attributes. Requirement attributes help define key information about the requirement. Each attribute forms a column on the traceability matrix.

The types and number of attributes selected should be appropriate for the needs of the organization and the chosen project life cycle. For example, unless the nature of the project or the organizational policies and processes require its use, a project using an adaptive life cycle may not require the use of a traceability matrix, because a less formal method is generally selected for tracing requirements. Projects following an adaptive life cycle may also collect fewer attributes. Other factors, such as the organizational culture may also influence the formality of the traceability approach. More entrepreneurial organizations may choose a less formal traceability approach.

For projects that use a traceability matrix, the business analyst may leverage the organization's standard traceability matrix template. When there is no traceability matrix template, then the matrix needs to be created as part of business analysis planning.

5.2.3.1 Requirements Attributes

For formal traceability, the business analyst should exercise care when selecting requirements attributes to ensure that the proper type and number are chosen. The business analyst should be sure to choose attributes that will actually be tracked and managed and not overlooked. Care should be taken to ensure that selected attributes do not overlap with information that is tracked and managed elsewhere. When information is duplicated, steps should be taken to ensure it is in sync in all locations.

The organization may determine that some attributes should always be considered, for example, creation date, last revision date, and version number, due to the value these attributes provide in managing and controlling requirement changes.

From the perspective of the *PMBOK® Guide* – Fifth Edition, typical attributes used in the requirements traceability matrix include but are not limited to:

- Requirement ID, which uniquely identifies each requirement;
- Short textual description of the requirement;
- Objectives:
  - Business need,
  - Business goals and objectives, and
  - Project objectives;
- Product development stage:
  - Design,

  - Build,
  - Test,
  - Implement, and
  - Verify;
- WBS (work breakdown structure), a cross reference to deliverables as identified in the WBS;
- Status, such as active, approved, deferred, canceled, added;
- Rationale for inclusion (why the requirement is important to include);
- Priority (how important the requirement is);
- Owner;
- Source (where the requirement came from);
- Version;
- Date completed;
- Stakeholder satisfaction;
- Stability;
- Complexity; and
- Acceptance criteria.

Organizations commonly use requirements management tools or spreadsheets to trace requirements. Figure 5-1 provides an example of a traceability matrix with attributes from the *PMBOK® Guide* – Fifth Edition. For some organizations, the traceability matrix that is kept on a spreadsheet becomes the de facto requirements repository.

	A	B	C	D	E	F	G	H	I
1	**REQUIREMENTS TRACEABILITY MATRIX**								
2	Project Name:		<optional>						
3	Cost Center:		<required>						
4	Project Description:		<required>						
5	*ID*	*Associate ID*	*Requirements Description*	*Business Needs, Opportunities, Goals, Objectives*	*Project Objectives*	*WBS Deliverables*	*Product Design*	*Product Development*	*Test Cases*
6	001	1.0							
7		1.1							
8		1.2							
9		1.2.1							
10	002	2.0							
11		2.1							
12		2.1.1							
13	003	3.0							
14		3.1							
15		3.2							
16	004	4.0							
17	005	5.0							
18									

Figure 5-1. Traceability Matrix with Attributes

5.2.3.2 Traceability Matrix Hierarchy

A formal traceability matrix is usually built hierarchically, starting with high-level requirements and filling in the details as the requirement is progressively elaborated. This hierarchy is similar to an outline that is filled in as more detail is known.

There are several advantages to creating a hierarchical grid:

- It provides a logical order for the requirements, one that is not disrupted when new requirements are added. It is clear where each new requirement belongs and whether it is related to another requirement. It also provides a way to organize requirements to ensure that there are no duplicate or conflicting requirements.
- Traceability can be started as soon as the first requirement is defined and detailed, because the requirements are progressively elaborated, allowing for the incremental development of the requirements.
- There is no implied sequence, which enables work to be easily divided up among the team.

The traceability matrix is built hierarchically; therefore, it provides for the evaluation of each new requirement. High-level requirements are evaluated to ensure alignment with their source. As high-level requirements are defined, these are included on the traceability matrix and traced to the business need, project objectives, and business goals. As lower-level requirements are defined, these are evaluated against the higher-level requirements for alignment. The lower-level requirements expand on the details articulated in the high-level requirements, and the progression is displayed easily on the traceability matrix. The matrix also helps to evaluate new requirements against associated high-level and lower-level requirements when they are added to ensure alignment in all directions.

> ***Collaboration Point***—Some organizations develop a traceability matrix template as a starting point. During business analysis planning, the business analyst collaborates with the project manager and business stakeholders to determine which components of the template are to be used during the project, which components complete the matrix, and at what point during business analysis is traceability considered to be complete.

5.3 Relationships and Dependencies

The traceability matrix is a tool that supports dependency analysis and impact analysis. Requirements are often related to other requirements; therefore, sometimes a requirement is not able to be satisfied in a solution without the other(s) being present. Dependency analysis is a technique used to discover these dependent relationships. Once analyzed, the set of requirements on the traceability matrix is recorded by grouping dependent requirements together. Some requirements management tools illustrate the dependencies visually by creating traceability trees.

Some examples of dependent relationships are as follows:

5.3.1 Subsets

A requirement may be a subset of another requirement.

Example—An organization may have different types of customers—retail customers and business customers—which are considered subsets or subtypes of a customer. All customers may have some common data, such as customer ID, name, and contact information, in addition to some common processes performed on the common data, such as ordering products. Each subset of customer may have data and processes unique to them. Retail customers may have a frequent buyer program and may allow customers to select preferred pickup times for products they purchase. Business customers may have a tax identification number and a line of credit, and may also be permitted to purchase products that are not available to retail customers. The hierarchical relationships of these requirements are easy to portray in a traceability matrix.

5.3.2 Implementation Dependency

Some requirements are dependent on the implementation of other requirements before they can be implemented.

Example—An organization is interested in developing a new sales report but discovers that some of the data to be included on the new report is not being captured. In this case, the reporting requirements are dependent on additional functional requirements to allow the new data elements to be captured on the customer order entry screen.

5.3.3 Benefit or Value Dependency

Sometimes the benefit of a requirement is unable to be realized unless another requirement is implemented first.

Example—An organization interested in improving wait times for customers calling into their phone reservation center finds out that a requirement to add a new phone line has been identified as a high-priority feature, but unfortunately the existing phone system is unable to accommodate any additional phone lines. This particular requirement's value will not be achieved until a new phone system is implemented.

5.4 Approving Requirements

Organizations and projects vary in how requirements are approved. Some organizations require a formal signoff on a requirements package, such as a business requirement document. In other organizations or for specific types of projects, the approval of requirements may be informal, requiring only a verbal approval.

The approval process is determined, documented, and approved up-front in business analysis planning. Determining the process early on helps to avoid conflicts later when the approvals are being sought. The approval process may include any of the following:

5.4.1 Work Authorization System

The work authorization system defines the process for authorizing work. It may include process steps, documents, tracking systems, and approval levels. The work authorization system is one of many organizational environmental factors that lie outside the control of the project team and therefore constrain the project team's options.

Example—The team prefers less formality, but is constrained by a formal approval process that has been recently introduced to solve the problem of scope creep. In agile/lean organizations, there is sometimes less formality about authorization, yet there is still some norm or informal agreement about what needs to be approved and how and when approval occurs. An agile team may agree that when the team wants to add any user stories, the product owner needs to approve them before they are put on the backlog, or it may decide that the team can add the items to the backlog, in which case the product owner's prioritization of the items becomes the mechanism for approval. Further approvals may be obtained during incremental demonstrations of the working product.

5.4.2 Approval Levels

Authorization levels are part of a work authorization system. Approval levels provide the detail regarding who has the authority to approve new and changed requirements. In absence of a work authorization system or when the current system does not cover requirements approval, the business analyst, project manager, and sponsor determine requirement approval levels in business analysis planning. Tools such as a RACI are helpful in facilitating discussions and decisions about approval levels.

There are different types of approvals. Distinguishing among these different types helps to avoid confusion about who gets to approve what. Some examples are:

- **Approval vs. signoff.** In one organization, business stakeholders may approve a set of requirements during a facilitated workshop and the sponsor may sign off on the requirements, knowing that the business provided verbal approval. In this example, there is a difference between approval and signoff. The sponsor is designated as the only person responsible for signoff. In another organization, the business analyst, the project manager, all business stakeholders, and the project sponsor are required to sign off on each business analysis deliverable including all requirements documents.
- **Reviewer vs. approver.** On one project, it was explained to the testers that they would review the requirements and their input was welcome, but they were not authorized to approve or reject requirements. On another project, testers review the requirements and when any are not clear, the testers have full veto power to reject any of the requirements.
- **Approval authority vs. accountability.** In one organization, business analysts were expected to be responsible for managing the requirements, the project manager was accountable for ensuring that the requirements were managed, and the sponsor was accountable for the end product and, therefore, the requirements of that product.
- **Rejection of requirements.** It is not always clear who can reject requirements. In some organizations, the power to reject requirements is granted only to those who are permitted to sign off on them. In other organizations, all participants in the requirements process may be provided with veto power, regardless of whether they have been granted approval authority or not.
- **Change Control Board (CCB) and approval of changes.** A CCB is a formally chartered group responsible for reviewing, evaluating, approving, delaying, or rejecting changes to the project, and for recording

and communicating such decisions. The CCB is often the ultimate source for approving requirements when there is a significant scope change beyond the project manager's ability to approve. Approved requirements are then baselined, and changes are proposed through the CCB.

The CCB may choose to have change requests under a prescribed dollar limit approved by business stakeholders and/or the sponsor, and change requests over the limit approved by the CCB. There are many ways for an organization to use a CCB within the requirements process. The process for the current project should be described and documented in the business analysis plan.

Collaboration Point—Organizations that use an adaptive life cycle may also have CCBs. When they exist, teams on adaptive life cycle projects and the CCB need to determine an optimal way to interact with each other.

- **Expert judgment and the approval process.** In addition to the business analyst, stakeholders may be asked to provide their expertise. Subject matter experts may be asked to sit on the CCB periodically or when specific changes arise based on their expertise relating to the proposed change. Their judgment and expertise are leveraged when assessing whether the change makes sense. When requirement changes require the review of one or more experts, this information should be documented in the business analysis plan. Knowing this detail up-front allows the business analyst and project manager to allocate sufficient time to evaluate requirement changes.

5.5 Baselining Approved Requirements

5.5.1 What is a Requirements Baseline?

The requirements baseline is the boundary that contains all of the approved requirements for the project, project phase, iteration, increment, release, or any other part of a project. The baseline provides a mechanism for comparison, thereby allowing the project team to recognize that a change has occurred. All approved work is inside the boundary or baseline. Everything outside the boundary needs to be approved. Once requirements are approved, these can be changed only through the change control procedures defined for the project.

The degree of formality applied to changes outside the baseline varies depending on the project life cycle and the organization's processes for managing change. A project using a predictive life cycle is apt to apply a more formal and complex change control process; a less formal process is used for projects with adaptive life cycles. See Section 5.5.3 (maintaining the product backlog) for more information regarding change in an adaptive life cycle.

5.5.2 Relationship of Requirements Baseline, Product Scope, and Project Scope

The project scope is the work performed to deliver a product, service, or result. The product scope is comprised of the features and functions that characterize the product, service, or result. Requirements describe features and functions of the final product, service, or result for the project; therefore, there is a direct relationship between the number of requirements and the product scope. The more approved requirements there are, the larger the product scope and the project scope.

5.5.3 Maintaining the Product Backlog

Business analysis work is important on projects regardless of whether the project follows adaptive or predictive life cycles. For projects that follow an adaptive life cycle, baselining requirements is performed by maintaining the product backlog. The product backlog is the list of requirements, usually written in the form of user stories. Although not commonly referred to as baselining, the principle is the same for all project life cycles. The backlog may have some large user stories, which when broken down, may include options that are not going to be selected and are not approved. On adaptive life cycle projects, a subset of the backlog is approved for each iteration.

The product owner is accountable for the product requirements, known as features, as well as for making priority decisions based on which requirements or user stories provide the greatest business value. Requested requirements (features) are written up as user stories and added to the product backlog throughout the project. New requirements often surface after a product review at the end of each iteration, but could be requested at any time.

The priority of user stories can change at any time. A requirement thought to be a high priority at the beginning of a project may be changed to a lower priority as the project progresses. On the other hand, the product owner could elevate the priority of other requirements that were originally thought to be unimportant. The business analyst or person performing that role may choose to maintain the status of requirements as the project, release, or iteration is executed. As requirements are added to the product backlog or changes in priority result in the movement of requirements from one release or iteration to another, the changes are tracked and communicated to the appropriate stakeholders as agreed upon in the communication plan, either formally or informally.

5.6 Monitoring Requirements Using a Traceability Matrix

Once the requirement attributes are determined, the traceability matrix created, and requirements approved and baselined, the requirements are monitored throughout the project life cycle. At this point, every detailed requirement should align with a business requirement. Business requirements are discussed in Section 2 on Needs Assessment.

Projects with an adaptive life cycle that use a traceability matrix may build the matrix incrementally in a consistent manner as details are elicited and analyzed.

5.6.1 Benefits of Using Traceability to Monitor Requirements

Monitoring requirements throughout the life cycle using a traceability matrix or similar structure helps to ensure that:

- More detailed requirements that surface are linked to the baselined requirements. As the requirements are progressively elaborated and additional details surface, the relationships are progressively elaborated. Every detail relates back to a higher-level requirement.

  Example—An organization has a baselined requirement to support multiple customer types. Through business analysis requirement workshops, it is uncovered that there is a difference in how complaints

are handled for each customer type. Detailed requirements emerge for a concierge service for business customers, which trace back to the initial requirement.

- A complete set of baselined requirements has the detail needed to build the feature or set of features. When a high-level requirement exists with no associated detail requirements, this points to an area needing further requirements elicitation to uncover the details.
- Supporting business analysis work products are created for each baselined requirement when needed.

  Example—An organization decides to create a concierge service for business customers. Process flows are developed to demonstrate the various interactions between the business customer and the concierge service. Training documents are developed to help staff address customer types and the associated processes.

- Work products created across phases, such as design and test documents, are created for each approved requirement.

  When there is a requirement with no associated work product, there is a risk that important components may be missed. For example, in a software development project, approved requirements need to be designed, built, tested, implemented, and verified, so that appropriate work products, such as the physical database design and other design documents, programs, and test cases can be developed.

  Requirement relationships are monitored to their related work products to help ensure that gaps between the approved requirement and what has been built and tested are not missed. When gaps are identified, the business analyst should:

  - Cancel or defer the approved requirement, or
  - Create the missing work product.

- Scope creep is prevented. PMI defines scope creep as the uncontrolled expansion to product or project scope without adjustments to time, costs, and resources. Projects using a predictive life cycle with many work products can monitor requirements relationships and also review each work product that is created throughout the project to ensure that it links back to an approved requirement.

  For example, when requirements are included on a traceability matrix, it is easier to see this linkage and spot missing requirements. By routinely comparing work products to requirements on the traceability matrix as they are created, gaps can be questioned and appropriate action taken.

  Example—A developer working on a user interface for an order entry screen decides to exceed customer expectations by highlighting the data fields that the customer changes to yellow to make them stand out. These changes are programmed without obtaining approval and are not discovered until user acceptance testing. The sponsor is not happy with the change, which ultimately results in rework, cost, and schedule overruns. Had test cases been created at the point of design or development, these changes could have been checked against the approved requirements and brought to the sponsor for approval. In this case, the request would have been declined because the sponsor knew that highlighted text is problematic for clients who use mobile devices to place orders.

Product owners on projects using an adaptive life cycle manage scope creep through regular reprioritization of the backlog. For adaptive projects, traceability matrices may be used to help manage scope creep or less formal techniques may be used to achieve the same level of thought.

5.7 The Requirements Life Cycle

The status or state of a requirement is a common attribute on the traceability matrix. The state of a requirement characterizes where it is in the requirements life cycle. The requirements life cycle, not to be confused with the project life cycle, represents the various phases that a requirement moves through as it is maintained across the project.

During business analysis planning, the list of requirement states is defined for the project. When an organization uses a requirements management tool, the requirement state is maintained automatically by the software after the business rules for determining state changes are configured within the tool. Without a requirements management tool, the business analyst maintains the states manually.

An authorizing body influences the requirement state when making decisions on requirements, such as:

- Approving a requirement but postponing it to another project, iteration, or project phase;
- Deferring a requirement to some unspecified future project, iteration, or project phase;
- Replacing an approved requirement that has not been started with another requirement;
- Canceling an approved requirement;
- Rejecting a requested requirement; or
- A combination of the above. For example, the authorizing body could approve one part of the requirement but defer another part.

Figure 5-2 is a state diagram that provides an example of some common requirement states and the life cycle that may be followed.

The actual states that a project tracks can vary considerably between, and sometimes within, organizations. For each project, within organizational constraints, a decision is made concerning which states the organization is willing and able to track and manage.

The requirement state is used when reporting requirement status to project stakeholders. For example, the state is required in order to report how many requirements are documented and how many are approved, deferred, or rejected in order to provide a fuller understanding of the conditions of the requirements on the project at any given point in the life cycle.

5.8 Managing Changes to Requirements

A change to a requirement may be proposed at any time during a project. Who can propose changes and how those changes are proposed are defined in the business analysis plan. Although suggested changes may be initiated verbally, these should be recorded for tracking and management purposes.

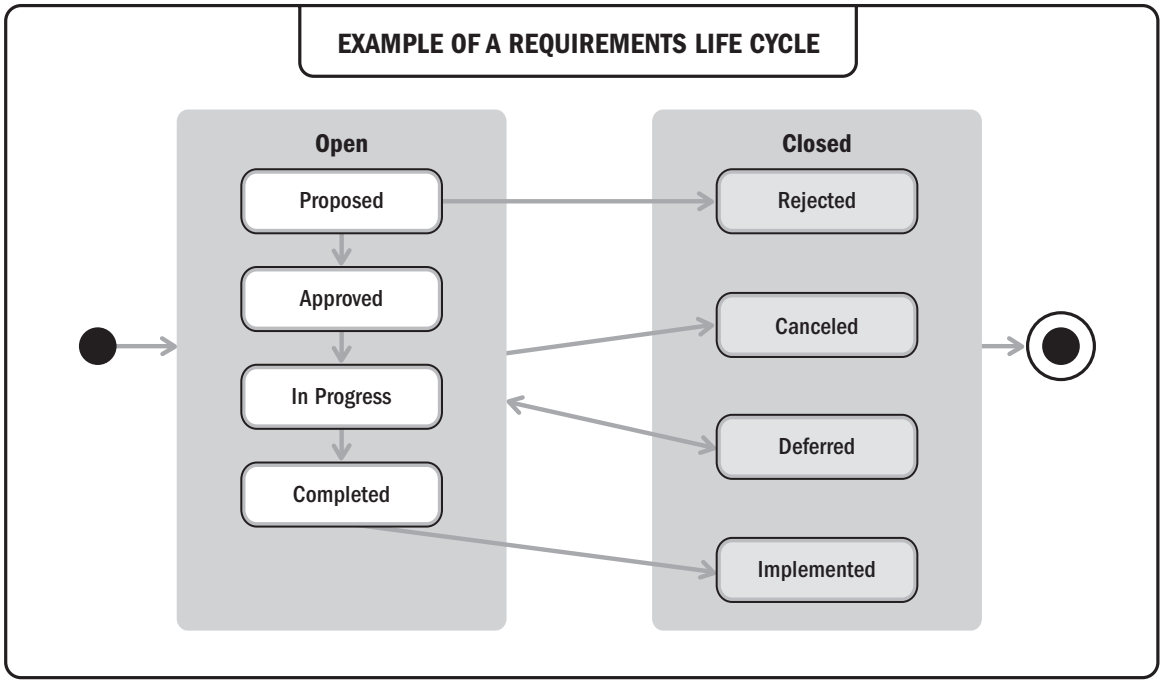


Figure 5-2. Example of a Requirements Life Cycle

Change requests are a part of the overall change control process. The *PMBOK® Guide* – Fifth Edition defines a change request as a formal proposal to modify any document, deliverable, or baseline. When the change request template is defined and maintained as an organizational process asset, the project team is required to use the approved template on the project. In cases where a template is not defined, the template needs to be created as part of business analysis planning. Some project life cycles use an informal change control process and, therefore, a less formal proposal is acceptable, for example, discussing the change.

The change control process defines how changes are submitted. For formal change control, change requests are submitted manually or as part of an online process. The business analyst monitors change requests and takes an active role in assessing the impacts of a proposed change. A change request process defines the information required to be gathered before a request can be considered for approval. The business analyst may be required to collect information based on the level of effort to make the change, cost impacts, risk impacts, and impacts on completed work or existing approved requirements. Eventually, every logged change request should be approved, deferred, or rejected by the change-approving body in accordance with the process defined in the business analysis plan. The change process may require a recommended turnaround time that the project team needs to adhere to when assessing a proposed change.

5.8.1 Change Management as it Relates to Business Analysis

Within change management, the role of the business analyst is to maintain the integrity of the requirements and the associated deliverables and work products and to ensure that each new requirement aligns with the business need and the business and project objectives. The key benefit of a requirements change process is to allow for

changes to documented requirements while reducing product and project risk, which often arises from making changes without consideration to the overall impact to the product and project, as well as considerations related to dependencies.

The applied level of change management is dependent upon many factors, such as the application area, the organizational culture, the organizational process assets, the complexity of the specific project and end result, contractual requirements, the project life cycle, and the context and environment in which the project is performed.

Project managers and business analysts both have an interest in change management. The project manager is concerned with all things associated to project scope. Approved change requests may require adjustments to the project management plan and other project documents, cost estimates, activity sequences, scheduled dates, resource requirements, and the analysis of risk response alternatives related to the project; therefore, the project manager is very involved in change management activities. The business analyst is interested in understanding the impacts to product scope, such as how the change request will impact documented requirements, existing and approved business analysis work products (e.g., process flows and models), and any product components that have been built and tested from existing approved requirements.

> ***Collaboration Point***—Assessing the impacts a proposed change will have on project and product scope requires the collaborative efforts of both the project manager and business analyst. It is important for the business analyst and project manager to have frequent communication about the proposed changes and their disposition.
>
> For projects using an adaptive life cycle, the team determines the details for most of the requirements on a just-in-time basis. Most details are elicited during the iteration or increment in which the requirement or user story will be delivered. Because requirements details emerge along the way, product owners and their teams expect changes during an adaptive project. The product owner regularly reprioritizes what is to be delivered, usually at the start of a new increment. Since adaptive projects rapidly deliver small increments, while changes tend to be frequent and each change is often small, it is more likely that small changes will be assessed on an informal basis. Changes that add value can be accepted and prioritized.

The approaches to change management taken by projects using adaptive life cycles are evolving. Variations for change management on adaptive projects include, but are not limited to, discussing changes and adjusting the backlog or Kanban board based on the discussion; managing change simultaneously from a long-range, mid-range, and immediate range perspective; or using automated tool support to streamline a more formal change management approach.

5.8.2 Change Control Tools and Techniques

Manual or automated tools may be used to manage change requests and the resulting decisions. These tools may already be in place within the organization. When a change control tool is being introduced on a project, the needs of all stakeholders involved in the change control process should be considered.

Collaboration Point—The business analyst and project manager need to work together to ensure that both business and technical stakeholders' needs are met when defining the change control process and the supporting tools.

5.8.2.1 Configuration Management System (CMS)

Configuration management helps to ensure the product or service being built conforms to its approved requirements. It provides a process to verify this conformance, document changes, and report the status of each change throughout the project life cycle. It includes documentation, a tracking process, and defined approval levels necessary for authorizing changes. It enables managing changes to aspects of a product in the context of the entire product as well as the context of other products on which it depends or which depend upon it.

Configuration management for business analysis ensures that (a) the requirements and requirement-related documents, such as, models, traceability matrix, and issues list, are stored where they can be easily accessed by project stakeholders and are safeguarded from loss; and (b) access to previous versions of documents is available, when needed. A business analyst may achieve these objectives with a CMS, a requirements management repository, or with a wiki platform. When using less formal tooling, such as a wiki, the CM goals will not be achieved without a process on how and when to access and write to the wiki. The process for storing requirements-related information and the tools to support the project needs are determined and agreed upon in business analysis planning.

5.8.2.2 Version Control System (VCS)

A version control system (VCS) tracks the history of revisions, but not always those that are related to software. A VCS is like a baseline in that the original code or work product is established, and changes to that code or work product are tracked. A VCS falls under the umbrella of a CMS and is one of the many processes that comprise configuration management.

Although often associated with software source code, a VCS can be used for managing business analysis documentation. When the requirements documentation is extensive, a VCS may be appropriate. Project teams need to decide whether or not previous versions of requirements and models are required. If this is a necessity for the project team, a VCS can help support this requirement.

Version control ranges from a simple system, for example, using the document's file name to reflect a date, time, and version number, to a more formal approach where documents are maintained by a tool that requires that documents be checked out of a library, locked by the system during the editing process, checked back in with mandatory comments explaining the changes made, and versioned automatically by the tool. When version control is required for the requirements documentation and when a version control system will be used should be noted in the business analysis plan.

5.8.3 Impact Analysis

When a requirement change is proposed, it is necessary to complete an impact analysis to evaluate the proposed change in relation to how it will affect other requirements, the product, the project, and the program. Impact

analysis is the work performed to assess a proposed change, which includes identifying the risks associated to the change, the work required to incorporate the change, and the schedule and cost implications. A key benefit of completing an impact analysis is that it allows for changes within the project to be considered in an integrated fashion, thereby reducing project and product risk, which often arise from changes made without consideration to the effect on the program, project, and end product.

Sections 5.8.3.1 through 5.8.3.5 describe a formal approach to impact analysis. Projects using an adaptive life cycle may use an informal approach to assess impacts and devise a course of action based on the value of the change along with its impacts. Impact analysis includes, but is not limited to, the following:

5.8.3.1 Impact on the Requirements Baseline

The business analyst reviews the change request to determine whether the request is a change to an existing requirement, a new requirement or set of requirements, or whether the change provides further detail of the same requirement. Using the traceability matrix, the business analyst identifies the requirements impacted by the change within the matrix. The business analyst can then quickly assess the affected relationships by impact, roughly quantifying how big or small and complex the change may be.

5.8.3.2 Impact on whether a Proposed Change Conflicts with Other Requirements

The business analyst assesses the proposed change against the baselined requirements or requirements residing in the requirements backlog. The business analyst is looking for situations where requirements could be in conflict with one another. When implemented, a requirement that is in conflict will cause another requirement to break or not be implementable. Conflicting requirements may break solution components that are already implemented. The business analyst analyzes the change request against the requirements baseline and existing solution components and notes areas of potential conflict.

When requirement conflicts arise, the business analyst facilitates a resolution to the conflict and may schedule a requirement session to discuss alternatives and reach consensus. It is very important to ensure that all impacted stakeholders are represented when sorting through conflicts. A proposed change request that is identified as being in conflict with existing product features, when approved, will overturn or override existing approved functionality. The business analyst has an important role to preserve the integrity of the requirements baseline, solution components, and existing features already implemented to ensure a proposed change adds the value that the organization expects.

5.8.3.3 Impact on Business Analysis

When conducting impact analysis, the business analyst assesses the impact that the proposed change will have on business analysis work that is currently in process, including work that has been completed and approved. The business analyst also considers how far along the solution is in the development cycle and works with the project team members accountable for product development to obtain input regarding how the proposed change, if approved, impacts the work completed to date.

The business analyst may need to replan current business analysis activities when the proposed change is deemed higher in priority than requirements that are currently in process. An adaptive project life cycle is used for some projects, which is well-suited for accommodating change.

Aside from assessing the impacts to the current business analysis activities, the business analyst considers any updates that are required to existing business analysis documents, because these deliverables are often the input to the project team members who perform work after the business analyst. These documents need to accurately reflect the requirements for the product and solution at all times in order for the development teams to produce the correct end product.

Interim documents or work products may need to be updated, but usually, these are created for a one-time use (e.g., a model that is built to elicit requirements from a specific stakeholder group). When the work products are not to be reused or referenced by project teams, then the business analyst does not need to spend time reviewing and revising them.

Process flows, use cases, business requirement documents, software requirements specifications, and user stories are examples of documents that may need to be revised whenever a change is approved. When the business analyst is conducting impact analysis, the time needed for revising the required business analysis documentation should be estimated in the event that the proposed change is approved.

5.8.3.4 Impact on Project Management

Even the smallest change needs to be understood in terms of the impact to the project. Approved changes may impact one or more subsidiary plans within the project management plan as well as in-process work that the project manager has been monitoring and controlling.

Maintenance of the project management subsidiary plans is outside the scope of business analysis, and not all plans may be impacted. The following list represents plans that a project manager may need to address whenever a change request is approved:

- Scope management plan,
- Business analysis plan,
- Schedule management plan,
- Cost management plan,
- Quality management plan,
- Process improvement plan,
- Human resource management plan,
- Communications management plan,
- Procurement management plan,
- Risk management plan,
- Stakeholder management plan,

- Scope baseline,
- Schedule baseline, and
- Cost baseline.

While adaptive projects tend to have fewer plans, they very often have a product roadmap, which shows, at a high level, what is planned for release over the course of the project iterations.

> ***Collaboration Point***—The project manager and business analyst can work together to assess project impacts. The time, cost, and risk impacts are reflected in the impact analysis to properly scope the level of effort associated with the proposed change. The project manager assesses project impacts, and the business analyst assesses impacts to the product. Product impacts may include changes to the business analysis deliverables required to build the end product, changes to scheduled business analysis activities, or changes to solution components in process or already implemented. To properly assess both the project and product impacts of a proposed change, the project manager and business analyst should work together, because each provide a perspective and level of knowledge that, when contributed jointly, will properly assess the size, cost, schedule, and risk impacts associated with the proposed change.

5.8.3.5 Recommending a Course of Action

After analyzing all of the impacts of the change and understanding the scope, risk, schedule, and cost implications, the business analyst assembles the results of the analysis. The business analyst recommends a course of action and includes this recommendation in the assessment. Some organizations look to the project team to provide more than one alternative along with the pros and cons, assumptions, risks, etc. for each alternative. Each impact assessment may result in an abbreviated form of a business case to speak to the value of the change. The purpose of the recommendation and supporting analysis is to provide those responsible for approving the change with all the information required to make a sound decision.

The process for approving changes was determined and approved up-front in business analysis planning. When the organization uses a CCB, the impact assessment is submitted for consideration. The business analyst adheres to the agreed change approval process for the project.

Once a decision is reached, the business analyst proceeds to address the outcome. The following courses of action are possible after the proposed change is evaluated:

- **Change approved.** The business analyst completes the necessary updates to the impacted business analysis deliverables. Planned and in-process business analysis activities are adjusted when impacted.
- **Change deferred.** The business analyst documents the decision along with a rationale for the decision. When a proposed date or product release is decided upon, this information is noted and reflected in the appropriate plans to ensure the change is addressed *at the requested future date.*
- **Change rejected.** The business analyst documents the decision and provides a rationale for the decision in the appropriate plan. Unlike the action for change deferred, there are no future date reminders established in the plans, because this work will not occur.

- **More information required.** Despite best efforts to ensure that the impact analysis is thoroughly constructed, sometimes the CCB or approval team requests more information. Conditions in the business may have changed since the project was first approved, or the CCB may now be considering different solutions or working from new data that was unavailable previously. This in turn involves another round of elicitation and analysis, an update to the impact analysis, and a resubmittal to the authoritative body considering the change.

Regardless of the decision made by the authorizing body, the business analyst has the responsibility of communicating the outcome of the change discussion with the project stakeholders. Interested stakeholders need to understand why a change was deferred or rejected and the rationale for the decision as much as they have a need to hear about the approved changed requests.

Projects using an adaptive life cycle often arrive at change decisions during incremental demonstrations of the working product. In iterative and adaptive projects, change is ongoing and emergent learning is often how the right product that maximizes value is identified. As in all projects, product owners in projects with an adaptive life cycle still need to think about the impact of the change on the product and project and to consider alternatives. Again, the degree of formality and the amount of documentation for the change decision process depends upon the requirements of the organizational policies and processes or external regulations.

5.8.4 Controlling Changes Related to Defects

Not all proposed changes are requests for new features or new requirements. Some changes are requests to fix defects that are identified in the solution. Such defects may be raised after formal or informal audits (e.g., inspections or walkthroughs) or may surface when stakeholders interact with the solution. Because a defect is a deviation from a requirement, the business analyst stays involved in the defect repair process by monitoring the repair or the replacement of the nonconforming solution component. Although the requirements-related documentation is not impacted, it is the component that is being modified to be aligned with the approved requirement. Defects in projects following an adaptive life cycle may be identified during incremental demonstrations of the working product and defect management may be informal. For more information on controlling changes related to defects, see Section 6 on Solution Evaluation.

6

SOLUTION EVALUATION

6.1 Overview of this Section

Evaluation consists of business analysis activities performed to validate a full solution—or a segment of a solution—that is about to be or has already been implemented. Evaluation determines how well a solution meets the business needs expressed by stakeholders, including delivering value to the customer. Some evaluation activities result in a qualitative or coarsely quantitative assessment of a solution. Conducting surveys or focus groups and analyzing the results of exploratory testing of functionality are examples of qualitative or coarsely quantitative evaluation. Other evaluation activities involve more precise, quantitative, explicit measurements. Comparisons between expected and actual results obtained from a solution are usually expressed quantitatively. For solutions involving software, analyzing comparisons between expected and actual values of data manipulated by the high-level functionality of the solution can be part of the evaluation. Nonfunctional characteristics of a solution (sometimes known as quality attributes) are often evaluated with measurements. For example, measurements are required to evaluate whether performance service-level agreements are being met. Additionally, comparing estimated and actual costs and benefits may be part of an evaluation of a solution.

This section of the practice guide covers both qualitative and quantitative evaluation activities. The techniques described herein are compatible with predictive, iterative, and adaptive project life cycles. Most of these techniques are useful for all kinds of projects and solutions. A few techniques are identified as specifically applying to solutions involving software.

Many of the techniques used during evaluation activities are also used during analysis or testing and sometimes during needs assessment. In addition, there is some overlap between evaluation techniques and traceability. The use of analysis techniques for multiple purposes is natural because one of the overarching goals of all analysis activities is to obtain a clear, unambiguous understanding of all aspects of a problem under consideration. This section of the practice guide refers to other sections in the practice guide when the activities and techniques were already described. Additional techniques are introduced as part of solution evaluation domain to precisely define acceptance criteria and enable measuring against the defined criteria. Techniques for defining acceptance criteria are also used as part of planning and analysis, as a basis for testing, and for defining service-level agreements and nonfunctional requirements.

6.2 Purpose of Solution Evaluation

Solution evaluation activities provide the ability to assess whether or not a solution has achieved the desired business result. Evaluation provides input to go/no-go business and technical decisions when releasing an

entire solution or a segment of it. For projects using iterative or adaptive life cycles, and for multiphase projects using predictive life cycles, evaluation may identify a point of diminishing returns. An example of this is when additional value could be obtained from a project, but the additional effort needed to achieve it is not justified. Evaluation of an implemented solution may also be used to identify new or changed requirements, which may lead to solution refinement or new solutions. Identification and definition of evaluation criteria also supports other analysis activities.

6.3 Recommended Mindset for Evaluation

6.3.1 Evaluate Early and Often

Evaluation is often associated with the end of predictive life cycles (e.g., user acceptance testing and release of a solution). For iterative or adaptive life cycles, evaluation may be associated with the end of a time segment (e.g., an iteration or sprint), or when a user story is delivered or completed for an adaptive approach that has no fixed time segments, such as Kanban. When evaluation criteria are defined early, they can be applied to a project in progress and provide input into the development of test strategies, test plans, and test cases. Early uses of lightweight evaluation techniques identify which areas of a solution need the most testing. Early articulation of evaluation criteria is an excellent way to specify or confirm both functional and nonfunctional requirements.

6.3.2 Treat Requirements Analysis, Traceability, Testing, and Evaluation as Complementary Activities

Adaptive life cycles explicitly define acceptance criteria with concrete examples as part of the elaboration of a user story. The acceptance criteria and the user story definition support each other. Together, these establish mutual agreement between business stakeholders and those responsible for developing the solution for what is required and how to know that the requirement has been met. Predictive and iterative life cycles also recognize the value of early testing in a project.

Early test specifications provide concrete examples that add clarity to requirements, regardless of how they are specified.

Formal traceability matrices provide verification that requirements support business goals and objectives and that testing sufficiently covers the requirements. Goals and objectives and the traces between them and the requirements that support them provide insight into candidate metrics for evaluation.

> ***Collaboration Point***—Business analysts, project managers, and testers should work together early on to create consistent and complementary approaches to analysis, testing, and evaluation activities. Testers can help check for completeness and consistency of all forms of requirements and evaluation criteria, and business analysts can help check for sufficient test coverage for areas of the solution that have the highest business priority.

6.3.3 Evaluate with the Context of Usage and Value in Mind

Validating a solution is more complex than validating to determine whether individual requirements have been met. Validation not only ensures that the solution is working as designed, but also confirms that it enables the usage and value that the business expected. No matter how requirements are elicited and specified, there is some intended usage associated with each requirement or group of requirements. The names of epics, user stories, and use case scenarios that focus on functionality provide direct identification of the intended usage. For epics and user stories that are not about functionality, usage is inferred from the "so I can..." part of the story, which provides its rationale.

> *Example*—The following epic is written for the insurance company discussed earlier in Section 2. It describes a need for information regarding the adjudication of claims. The "so I can" statement alludes to the usage of the data:
>
> "As an operations manager, I need data about the percentage of claims that were automatically adjudicated and the average time for a manual adjudication, so that I can assess how the revised adjudication practices are performing."

Nonfunctional requirements that define expectations for the performance or operation of a solution also have an implied context of usage, because a solution is expected to perform or be available, scalable, or usable so that work can be accomplished.

6.3.4 Confirm Expected Values for Software Solutions

For automated solutions, it is essential to validate functionality that manipulates data by using the solution's data retrieval functionality. However, the sufficiency of the solution also needs to be evaluated by confirming that the actual data is accurately stored, because that data may potentially be retrieved in other ways for other users and uses.

> *Example*—Consider the insurance company example, where the business need is to provide a software solution so that a claim can be submitted via a mobile device. Part of the evaluation of the solution is to look at the results of tests where a claimant submits a claim using the mobile device and also look at the results of tests where the claim is retrieved using the mobile device to ensure that the actual claim is presented as expected. However in an insurance company, a claim can be retrieved in other ways, such as through a reporting mechanism or another interface such as a claim adjuster's workstation. When those usages are not part of the scope of the solution, it is still part of the scope to accurately store the claim data so that it is available for any usage.
>
> To verify that the data is accurately stored and available for other forms of retrieval, the results that were obtained by accessing the data directly from its storage location should be reviewed to confirm that the actual stored values match the expected stored values. Even when direct confirmation of stored data is included during testing, it is worthwhile to confirm it again as part of evaluating the most-used

portions of the functionality. Evaluation of a software solution is usually conducted against a system that is either about to be released or has already been released; therefore, testing during evaluation usually encompasses a larger body of data than the initial testing. As a result, new and different kinds of data anomalies may be uncovered by looking at the results of tests used to directly confirm the data, and these anomalies will need to be analyzed.

6.4 Plan for Evaluation of the Solution

Undertaking evaluation activities is not a trivial task. It takes time and effort to identify and confirm evaluation criteria, implement anything necessary for taking measurements that is not already built into the solution or its infrastructure, take the measurements, and report on and analyze the measurements.

Factors to consider when planning for evaluation activities include:

- **What project or organizational goal, objective, or risk does this evaluation activity monitor or track or confirm?** Project implementations are often inundated with monitoring and tracking dashboards and reports that are never used or sometimes never even viewed. It is important to tie every evaluation activity and every individual metric to organizational or project priorities.
- **Who will cover costs for the time and effort needed to conduct evaluation?** When there are plans to evaluate the long-term performance of a solution, ongoing or periodic evaluation activities are needed for a period of time that is longer than an initial evaluation. Depending upon organizational responsibilities, evaluation activities may be charged to a project or to quality assurance or some other part of the organization. The organizational area that is going to assume the costs for evaluation should be involved in approving the estimates for the evaluation activities. These estimates should be provided as early as possible in the project life cycles.
- **Does the solution or its infrastructure have built-in measurement capabilities for the evaluation criteria?** Additional measurement capabilities may be needed when none are built-in or when what is built-in is not sufficient. For software solutions, additional data may need to be stored programmatically to enable measurement. Every additional data element that is needed should be recorded either manually or automatically (programmatically); each additional data element requires either a human procedure or software code to ensure that it is captured.

  Example—One of the goals in the insurance company example was to reduce the amount of time it takes to process claims. When there are capabilities to capture the date and time a claim was filed and the date and the time its adjudication was completed, then it is already possible to determine the elapsed time to process a claim. Elapsed time may be a sufficient evaluation criterion; however, when a further breakdown is needed, additional data may need to be captured (e.g., how long it took to submit the claim, determine eligibility, or adjudicate the claim; whether or not the claim was in a pending state and, if so, for how long; how often were eligibility and adjudication automated vs. performed manually by a claims adjuster; or the date when payment finally arrived in the claimant's mailbox or bank account).

- **Are there already ways to extract measurement data to use in the evaluation?** When data exists, it does not necessarily mean that it is available for evaluation.

  Example—Looking at the insurance company, if the accepted quantitative measurement for length of time to process a claim is the elapsed time between the date and time a claim was filed to the date and time it was adjudicated, is there already a software query or report or a human procedure that reviews all claims and calculates the average elapsed time to process a claim, organized by some rolling duration? If not, someone is going to have to write the query or create the report or create a human procedure to make the measurement available or perform the required calculation to support the evaluation. Ways to extract and report on qualitative or semiquantitative evaluation data need to be available or created. In addition, if the insurance company decides to use a customer satisfaction survey as part of how it evaluates its new mobile app for claims submission, the results of that survey need to be aggregated and analyzed.

- **Is the chosen evaluation method effective and relatively inexpensive?** Organizations that attempt to evaluate solutions and project outcomes sometimes discover that the evaluation activities are extremely costly. This is particularly true when additional data—above and beyond the data needed for the solution—needs to be embedded in the solution in order to evaluate it.

  Example—Using the insurance example, if it is necessary to evaluate whether all populations can easily use the new mobile application for home and auto insurance claims, demographic data that is not required to process the claims needs to be collected. Adding and maintaining data elements about the claimant, which are not core to the solution, has an added cost that some businesses may not want to pay for. With that in mind, the insurance company may decide to collect this information for its own evaluation in a less precise, less expensive way that is still effective. The insurance company could conduct a survey with questions about the usability of the new application and also include some demographic questions to segment the responses. The survey will not have the precision to tie specific demographics directly to a specific claim that was submitted, but it will be sufficient for the purposes of evaluating the usability of the solution.

- **Are there already existing ways to report and publish the results of an evaluation?** Some organizations already have home-grown or purchased automated tools that produce dashboards or scheduled reports or use report templates when evaluations are conducted manually. If these kinds of tools or templates are not available or if the evaluation requires data that has not been previously captured, then there will be additional costs incurred to make the results of an evaluation available.

6.5 Determine What to Evaluate

There are a number of factors to think about when identifying evaluation criteria. This section presents a list of these items for consideration. Section 6.6 provides information on techniques for conducting evaluations and for defining evaluation criteria and acceptance levels.

6.5.1 Consider the Business Goals and Objectives

Business goals and objectives and priorities enable a project to launch and hopefully result in a solution. The solution needs to be evaluated against those goals and objectives. In this practice guide, Section 2 on Needs Assessment emphasized that goals and objectives need to be SMART. The measurements specified in the goals and objectives serve as good clues as to what needs to be measured during the evaluation of the solution.

6.5.2 Consider Key Performance Indicators

Key performance indicators (KPIs) are metrics that are usually defined by an organization's executives. These indicators can be used to evaluate an organization's progress in achieving its objectives or goals. Typical broad categories of KPIs are: finance, customer, sales and marketing, operational processes, employee, and environmental/corporate, and social responsibility/sustainability. Sometimes information technology is rolled into the operational processes category and sometimes it is considered a separate KPI category.

Most project goals and objectives are associated with one or more of these KPIs. For organizations that already define and measure KPIs, one or more of these KPIs can be used to evaluate the solution, taking advantage of measurement capabilities for KPIs that have already been used in some way.

6.5.3 Consider Additional Evaluation Metrics and Evaluation Acceptance Criteria

While many of the metrics and acceptance criteria for evaluating the solution fall out of the goals, objectives, or KPIs, there may be additional metrics and acceptance criteria to consider. Some examples of additional metrics and acceptance criteria are:

6.5.3.1 Project Metrics as Input to the Evaluation of the Solution

When considering project metrics as input to the evaluation of a solution, it is important to distinguish between those metrics that support evaluating the solution and those that focus on project execution.

Actual project costs can be inputs to financial evaluations of a solution, such as a recalculation of return on investment or net present value where projected costs are replaced by actual costs and benefits are revised when they are better known.

Metrics derived from measurements of cost, effort, and duration, such as variances between estimated amounts and actual values, are used to track project progress and performance. Some organizations use earned value calculations for this purpose. While variances and earned values are important metrics for many organizations when evaluating projects, they do not explicitly evaluate the solution, and in any event, are usually within the purview of project management rather than business analysis.

Tracking change requests can be used as an indicator of project volatility but not as an indicator of solution viability. Tracking the status of requirements is a way to track project progress, but may or may not reflect the

completeness of the solution or how well it serves its purpose. Tracking the number of defects identified and the number that have been fixed is indicative of effort undertaken to address quality; however, it does not reflect the actual quality of the solution nor whether it provides value to the stakeholders or customers.

Projects using an adaptive life cycle, such as Scrum or Kanban, often use other metrics to reflect a project team's rate of progress on a project. The measurements are often expressed in terms of burndown (the number of backlog items remaining at any point in time) and velocity (the number of backlog items completed during a delivery interval). These measurements reflect the efficiency of the project, what the project has delivered, what was deliberately not delivered (descoped), and what remains to be delivered. Again, these are primarily project execution metrics. However, these metrics can contribute indirectly to evaluating a solution. Minimally, at a qualitative level, agile metrics may be assessed as part of evaluating whether the solution was delivered to the business faster and whether delivering some functionality sooner provided the business with any benefits (e.g., an overall increased market share or a lift in the amount of business conducted with existing customers).

> ***Collaboration Point***—Business analysts and project managers should work together with stakeholders to consider which project metrics should be incorporated into evaluation activities.

6.5.3.2 Customer Metrics

From a customer perspective, evaluation sometimes focuses on qualitative aspects, such as satisfaction, but even qualitative aspects can be measured semiquantitatively.

> *Example*—Using the insurance example, what percentage of customers responding to the survey report indicate they are very pleased or extremely pleased with the new mobile app? How many calls does customer support receive about the new mobile app? What is the trend in the frequency of the calls? How many of these calls are complaints? Evaluation from a customer perspective can also be more quantitative. For example, when assessing the usability of the insurance mobile app, if the business expected that customers could file 90% of their own claims in less than 10 minutes, is this occurring?

6.5.3.3 Sales and Marketing Metrics

Sales and marketing may have ranges of measurable goals for the project (e.g., a range of expected values for overall increased market share or a range for percentage increase in the amount of business done with existing customers). The solution can be evaluated to determine whether or not these expectations have been met.

6.5.3.4 Operational Metrics and Assessments

Operational metrics may be functional or nonfunctional and can be considered from a systems perspective, a human perspective, or both. For organizations that define and measure operational KPIs, it may be possible to reuse one or more of the metrics to evaluate the solution, provided it is possible to determine the impact of the solution on the KPI.

As noted in Section 4 on Requirements Elicitation and Analysis, process flows (swimlane diagrams) can be annotated, either on an overall basis or a stepwise basis, with baseline, target, or actual key performance indicator (KPI) metrics.

6.5.3.5 Functionality

Sections 6.6.1 through 6.6.7 describe some techniques that are used to evaluate functionality, such as reviewing the results of verification activities for specific broad business usages of the solution, as represented by bunches of use cases or scenarios or users stories or functional requirements executed in a natural sequence. With the exception of situations where errors in the solution could result in an unacceptable level of risk to life, property, or financial solvency, evaluation should not require a review of the results of the individual tests conducted during the development of the solution, other than to confirm that the test coverage was sufficient.

Nonfunctional requirements are used to specify overall system-wide characteristics of the solution, such as performance, throughput, availability, reliability, scalability, flexibility, and usability.

> *Example*—What is the availability of the new mobile application? Can it support an x% increase in the number of users using it simultaneously (where "x" is defined by the business)?

Organizations that already define and measure information technology KPIs may already have instrumentation to measure system-wide, nonfunctional requirements, such as availability. Nonfunctional requirements can also be evaluated at a usage level.

> *Example*—In the insurance example, from a usage perspective, how easy was it for a customer to file a claim (where "easy" is defined in a measurable way)? How fast did an automatic adjudication occur?

Clearly specified nonfunctional requirements should be written to identify their measurable acceptance criteria. Refer to Section 6.6.6 for ways to define measurable acceptance criteria for nonfunctional requirements. Measurable acceptance criteria can also be the basis of service-level agreements.

6.5.4 Confirm that the Organization Can Continue with Evaluation

As measurements are identified, review any estimates for the costs of evaluation and update them as needed. When necessary, reconfirm that the organization or department who is sponsoring the evaluation and covering costs is still able or willing to do so.

6.6 When and How to Validate Solution Results

For a predictive project life cycle, validate the solution at the end of the project life cycle either immediately before release or at an agreed-upon time after release.

For an iterative or adaptive project life cycle, validation is performed at the end of every iteration, sprint, or release, when the team provides production-ready functionality for the stakeholders to evaluate.

The following evaluation techniques can be used to evaluate solution results:

- Surveys and focus groups,
- Results from exploratory testing and user acceptance testing,
- Results from day-in-the-life (DITL) testing,
- Results from integration testing,
- Expected vs. actual results for functionality,
- Expected vs. actual results for nonfunctional requirements, and
- Outcome measurements and financial calculation of benefits.

While this list is not exhaustive, it does contain widely used techniques. These techniques are described in more detail in Sections 6.6.1 through 6.6.7.

6.6.1 Surveys and Focus Groups

As previously mentioned in Section 4 on Requirements Elicitation and Analysis, surveys can be used to gather information and elicit requirements from a very large and/or geographically dispersed population. Survey questions can solicit qualitative or semiquantitative feedback about satisfaction with the solution, how it is performing, or what aspects of the solution present challenges to its users. All of the advantages and concerns mentioned in the analysis section of this practice guide also apply to the use of surveys for evaluation.

Focus groups provide an opportunity for individuals to offer thoughts and ideas about a topic in a group setting and to discuss or qualify comments from other participants. Like surveys, focus groups are useful either as an elicitation technique or as an evaluation technique to obtain feedback from stakeholders in general and, more specifically, from users of the solution.

6.6.2 Results from Exploratory Testing and User Acceptance Testing

Exploratory testing is an unscripted, free-form validation or evaluation activity conducted by someone with in-depth business or testing knowledge. Generally, exploratory testing should be conducted in addition to (not in place of) formal testing. User acceptance testing is a formal testing activity which validates that the solution meets the defined acceptance criteria. It is conducted by someone with in-depth business knowledge.

For evaluation, the results obtained from exploratory testing and user acceptance testing are used to determine whether or not a product, service, or solution is working as intended with regard to functionality and ease of use, and when applicable, performance. When done well, exploratory testing also attempts to uncover whether or not a product, service, or solution responds properly to unintended uses (does not permit them) or whether it can be used in ways that were not intended.

6.6.3 Results from Day-in-the-Life (DITL) Testing

DITL testing is a semiformal activity, conducted by someone with in-depth business knowledge. For software solutions, DITL testing consists of a set of use case scenarios or several user stories or functional requirements to exercise in sequence against a specific segment of data, in order to compare the expected results with the actual results. Sometimes day-in-the-life testing is used more broadly as a semiscripted form of exploratory testing.

For evaluation, the results obtained from DITL testing help to determine whether or not a product, service, or solution provides the functionality for a typical day of usage by a role that interacts with the solution.

> *Example*—Using the insurance example, the solution can be evaluated from the DITL perspective of a claims submitter or claims adjuster.

6.6.4 Results from Integration Testing

Integration testing is used to validate or evaluate whether a solution does what is expected in the larger context of other ongoing business and systems operations in the organization. From the perspective of software solutions, integration testing is more encompassing than systems testing. Systems testing is a form of verification that proves when a solution being developed can interoperate with other systems and organizations that request services from it or vice versa. Integration testing places the solution in an environment that is either identical to or nearly identical to the production environment. For software solutions, organizations that have the resources to maintain an isolated separate production-like test environment are able to conduct integration testing prior to the release of the solution. Organizations with smaller test environments can conduct an integrated evaluation in their preproduction environment.

By examining the results of integration testing, it is possible to evaluate a solution in the larger context, because new and different kinds of anomalies may be uncovered when a solution is introduced into a production environment.

6.6.5 Expected vs. Actual Results for Functionality

This section provides a generic way to capture acceptance criteria and expected vs. actual results for functionality. During requirements elicitation, broad acceptance criteria are sometimes defined in terms of the actual usage of a solution. Acceptance criteria defined as part of evaluation need to be broad, yet still have expected results that are compared to the actual results. A DITL test, such as a day-in-the-life of a claim submitter or claims adjuster, can be constructed from one or more functional acceptance criteria.

One format for defining functional acceptance criteria is presented in Table 6-1 followed by a definition for each of its terms and a sample of broad acceptance criteria, using the insurance example.

Note that this format is independent of whether or not the solution is automated, in whole or in part.

Table 6-1. Sample Format for Defining Functional Acceptance Criteria

Field	Definition	Sample from the Insurance Example
Preconditions	Whatever needs to be true within the evaluation boundary to evaluate against the acceptance criteria	• Active medium-term customer of the insurance company • The claim's maximum automatic reimbursement amount for automatic adjudication is defined • Service-level agreement for manual claims adjudication is defined
Event	The specific action that is to occur, along with any specific input data needed for the action	Submit a claim that exceeds the claim maximum automatic reimbursement amount
Expected Result	A list of the expected responses, which may be a response message or acknowledgement and/or output data and/or post-conditions that may be observed.	• Claim is pended • Customer receives a message that the claim will be adjudicated by a claims adjuster within the number of days specified by the service-level agreement for manual claims adjudication

Functional acceptance criteria may be tabulated as shown in Table 6-1 or by placing all of the fields in one row, thereby making each functional acceptance criterion a row entry below the field names. Many business analysts add an additional field for actual results to aid in the analysis of expected vs. actuals (see Table 6-2).

Table 6-2. Sample Format for Analyzing Expected vs. Actual Results

Preconditions	Event	Expected Result	Actual Results
Customer John Jones is an active medium-term customer Claim's maximum automatic adjudication amount is $500 Manual adjudication service-level agreement is 5 days	John Jones submits a claim for $600	Claim is created as a pended claim Message is "Thank you for submitting your claim. Your claim will be forwarded to a claims adjuster and will be adjudicated in 5 days."	
Preconditions for another event			

6.6.6 Expected vs. Actual Results for Nonfunctional Requirements

This section provides a generic way to define acceptance criteria and capture expected vs. actual results for nonfunctional requirements. When the acceptance criteria for nonfunctional requirements are defined quantitatively within reasonable ranges, these criteria can also serve as the basis for service-level agreements.

The optional format shown in Table 6-3 can be used to define nonfunctional acceptance criteria. Each of the terms is followed by a definition and a sample of acceptance criteria, using the insurance example.

Table 6-3. Sample Format for Analyzing Expected vs. Actual Results for Nonfunctional Requirements

Note: This format is applicable for both manual and automated solutions (claims submitted on paper or claims submitted online).

Field	Definition	Sample from the Insurance Example
Type	Designation of the kind of nonfunctional requirement for which the acceptance criteria is being defined	Usability
Name	A unique designation of this nonfunctional requirement	Customer ease of use
Description	A very short summary that explains what is to be measured	Elapsed time between the time a new customer starts to submit a claim and the time that the new customer completes the submission (a new customer is someone who is submitting a claim for the first time)
How-to-measure	How the solution characteristic will be measured and what the measurement units will be	Elapsed time in minutes to complete a claims submission (as measured by a stopwatch for paper forms and by weblog timestamps for mobile app)
Worst-case value	The minimally acceptable value	Not more than 15 minutes for 90% of the measurements
Target value	The expected value for the solution characteristic	Not more than 10 minutes for 90% of the measurements
Best-case value	The ideal measurement for the solution characteristic, where there is no commitment to achieve this value	Not more than 5 minutes for all measurements

6.6.7 Outcome Measurements and Financial Calculation of Benefits

Some expected benefits for a project are readily tied to factors that can be measured; others may need to be derived or inferred.

Example—Using the insurance example, one of the benefits was to decrease the time to process a claim. The average time to process a claim should be measurable and the solution should be evaluated as to how well it was achieved. Beyond the actual measurement, as part of evaluation, it may be important for claims operations to derive the financial benefit of decreasing the time to process a claim. The example shown in Table 6-4 represents one way in which the derivation of such a benefit may be specified.

Table 6-4. Sample Outcome Measurement and Financial Calculation of Benefit

Topic	Claims Adjuster Productivity
Name of outcome	Decreased time to process a manual claim.
How measured	Difference between the previous and current average number of minutes for a submitted claim to be paid. (All claims were manual prior to the implementation of the solution; now, a manual claim is one that cannot be adjudicated automatically using the solution developed by the project.)
Inferred/derived financial benefit	(Total number of hours that claims adjusters previously spent on manual claims per calendar quarter) minus (total number of hours that claims adjusters currently spend on manual claims per calendar quarter) multiplied by (average hourly salary of a claims adjuster).
Sources of record	Claim adjuster timekeeping (timesheets); payroll.
Importance	This outcome is needed to evaluate operational productivity. The inferred/derived financial benefit is needed to evaluate actual costs/benefits against expected costs/benefits.
Current ability to measure	The needed information is available; a business decision is needed for how/whether to include overtime in the calculation.
How fresh does info need to be	Most recent quarter and previous quarter.
How trended/tracked	Quarter to quarter comparison.
Areas which need this metric	Claims operations.

In the claims example in Table 6-4, the information needed for the calculation is readily available. In situations where it is not, the business should decide whether it is willing to spend the money for the capability to collect the information needed to conduct the evaluation.

6.7 Evaluate Acceptance Criteria and Address Defects

As evaluations occur, regardless of the type of life cycle, one or more of the following activities in Sections 6.7.1 through 6.7.3 occurs.

6.7.1 Comparison of Expected vs. Actual Results

For functional or nonfunctional requirements or user stories or use cases with defined acceptance criteria, the expected results can be compared with the actual results.

> *Example*—In the insurance example, the worst-case value and target values defined in the acceptance criteria for customer ease of use indicate values that can be compared with the actual ease of use. This comparison may also be accomplished for qualitative evaluations, such as surveys, where worst-case and target values provide a way to compare expected with actual values. For example, the minimum for customer satisfaction could be that 70% of the survey respondents indicate that they were very pleased or extremely pleased with the new mobile app; the target could be 80% of the respondents rated their satisfaction as very pleased or extremely pleased. These values are then compared with the actual responses.

6.7.2 Examine Tolerance Ranges and Exact Numbers

Evaluations based on an exact-value prediction of expected results have often been conducted in project management by comparing estimated costs and efforts to actual values and providing an explanation or reason for the variances. For acceptance criteria for nonfunctional requirements, expected values take the form of tolerance ranges, where the worst-case value corresponds to the minimum acceptable value, the best-case value corresponds to the wished-for value, and the most likely value corresponds to the target value. Thus, the expected value range is from worst-case to best-case, with the target lying somewhere in between the two. Using ranges as evaluation criteria for comparing expected values to actual results provides a way to recognize the inherent real uncertainties involved in developing a solution. At the same time, the minimum acceptable value represents a commitment by the project team to remain mindful of business needs. If the actual solution delivers less than the minimum acceptable value, the solution is defective in that regard.

6.7.3 Log and Address Defects

When the actual results of an evaluation for a solution do not match the expected results, it is important to analyze the cause for the discrepancy. Sometimes evaluations uncover defects that were not previously found, and these should be logged and assigned for resolution. Sometimes evaluations identify that a service-level agreement is not being met, and these situations should also be logged and assigned. Depending on the organization, defect logging and tracking can be performed on an informal or formal basis.

Evaluation may discover or confirm that the actual values do not meet the expected goals, and the reasons may or may not be something that the solution is expected to accommodate.

> *Example*—The insurance company expected 80% of the claims to be adjudicated automatically as a result of the solution, yet only 50% of the claims were automatically adjudicated. The reason may be that the business did not define a sufficient number of rules for automated adjudication in order to reach the goal, or it may be that there is a defect that is sending claims to the adjusters when the claims should be automatically adjudicated. It is also possible that a recent natural disaster caused a spike in larger and more complex claims.

As with all defects, when a defect is found during evaluation, it should be addressed by additional requirements analysis, impact analysis, and change requests, as applicable.

6.8 Facilitate the Go/No-Go Decision

When evaluation takes place prior to the release of a full solution or a segment of a solution, stakeholders need an opportunity to decide whether or not the solution should be released in whole or in part or not released at all. The stakeholders in the RACI matrix who were identified as having a role to approve or sign off on the solution are generally the individuals who make the go/no-go decision. It is important to summarize the evaluation results in a meaningful way, because evaluations can produce voluminous amounts of information. Whenever possible, the evaluation results should be presented in tabular or visual form (i.e., charts/graphs/pictorial) to help decision makers grasp the impact and render a decision.

Whenever possible, go/no-go decisions should be made during an in-person meeting to allow all stakeholders to hear the rationale for the decisions from their counterparts. Like any well-run meeting, the use of a time-boxed agenda that is shared with all of the stakeholders prior to the meeting is recommended for go/no-go decision meetings. The stakeholders should be provided with access to the summarized evaluation results prior to the meeting. An agreed-upon decision model for how to reach a decision should already be in place. A “go” permits the release of the solution. A “no-go” either delays or disapproves the release of the solution.

The individual who analyzed the actual vs. expected results should attend the decision meeting. In some organizations, that individual also facilitates the meeting, although the role that facilitates this meeting depends on organizational preferences. In any event, the individual who conducts the meeting should be experienced as a facilitator.

Sometimes, an evaluation result reveals an overriding reason to delay the release.

> *Example*—In the insurance example, when the solution exhibits very poor scalability, it may be beneficial to delay the release until the scalability defects are addressed. When a new regulatory constraint emerges, which requires a new adjudication calculation and results in much smaller or much larger payments per claim than provided by the solution, it may be wise to delay the solution until the calculation is adjusted.

Sometimes an evaluation reveals a “show stopper,” which forces a no-go decision. Such decisions can occur in projects that present extreme risks for life or property or large-scale finances.

6.9 Obtain Signoff of the Solution

The RACI developed previously in Section 3 on Business Analysis Planning designates who is accountable for signoff on the solution. The formality of signoff depends upon the type of project, the type of product, the project life cycle, and corporate and regulatory constraints. For example, formal signoffs for projects are a common protocol when one or more of the following characteristics are present:

- Projects with a line of business-wide or enterprise-wide impact;
- Products where errors or failure to achieve tolerances could result in death or an unacceptable level of risk to life, property, or financial solvency;
- Projects in organizations following strict predictive life cycles; and
- Heavily regulated industries, such as banking and insurance or medical devices, clinical research, or pharmaceutical companies.

Organizations with any kind of formal signoff should reach agreement about the format of a signoff document, how it should be recorded, how it should be stored, and whether signoff needs to be obtained when all the signatories are physically together or whether it is acceptable to sign remotely. For example, some organizations require "wet signatures" (handwritten signatures) for signoffs, while others allow electronic signatures, email correspondence, or electronic voting.

Organizations with informal signoff practices need to obtain signoff in the manner that is acceptable to the organization.

> ***Collaboration Point***—The project manager and the business analyst, working with auditing and project stakeholders, should work together to determine the signoff practices required by the organization and for the project.

For a predictive project life cycle, signoff often occurs at the end of the project life cycle either immediately prerelease or post-release. For an iterative or adaptive project life cycle, informal signoff generally occurs at the end of each sprint or iteration, with formal signoff occurring prior to release of the solution.

6.10 Evaluate the Long-Term Performance of the Solution

Evaluating long-term performance is part of assessing the business benefits realized by implementing the solution. Long-term performance is a broad concept and applies to the execution or accomplishment of work in any form by people or systems or both. Almost anything in a solution that was identified for measurement prior to release can be evaluated at regular intervals over the long-term to identify positive or negative performance trends. For businesses that make use of business intelligence techniques, once a solution is released, any and all information captured as part of that solution can be analyzed to identify accomplishments and trends. Modern marketing organizations rely on analytics/business intelligence capabilities to evaluate whether or not marketing campaigns achieved the desired changes in customer behavior in the long-term.

Section 6.6.7 illustrates a way to evaluate claims adjuster productivity. That evaluation could continue on a quarterly basis to see whether or not the solution continues to deliver increases in productivity or whether a plateau is reached.

> *Example*—In the insurance example, analytics can be used to evaluate whether a business rule that was created to automate some of the claim payments is contributing to increased or decreased overall costs for the company.

6.10.1 Determine Metrics

When evaluation is planned (see Section 6.4), the key metrics are identified well before evaluation takes place. Business intelligence/analytics applied to the information captured by the solution may reveal additional metrics. Additionally, post-release, stakeholders may identify new metrics that tie to the current or new objectives.

When collected on a regular, periodic basis, the metrics noted in Section 6.5 can be used to evaluate long-term performance. Businesses with established KPIs often use the metrics associated with them as an input to long-term evaluation of performance. Organizations with more modest performance measurement capabilities may prioritize a subset of metrics to use for long-term evaluation.

> ***Collaboration Point***—Project managers and business analysts should work together with stakeholders to identify and prioritize which performance aspects should be monitored over the long-term and to confirm that the business is willing to cover the costs for the continued measurements and evaluation.

6.10.2 Obtain Metrics/Measure Performance

Realistic planning for evaluation (see Section 6.4) is essential for obtaining metrics to measure performance. It can be costly to obtain metrics when the capability to capture them is not already present in the organization and its infrastructure, prior to the release of the solution, or when it is not built into the solution as part of its development.

Usually, the same techniques selected to obtain, extract, and report on metrics during solution evaluation can be used to obtain metrics to measure long-term performance. Measuring long-term performance requires comparing individual and cumulative metrics collected during two or more comparable time periods (e.g., current year vs. prior year, or this quarter vs. last quarter vs. the quarter prior to last quarter).

6.10.3 Analyze Results

Analysis of long-term performance examines the measurements, identifies stable metrics (plateaus), and metrics that are trending upward or downward, and discovers anomalies in the measured values or trends. Once the behavior of the metrics is known, further research and root cause analysis can identify the problem that is adversely affecting performance.

Example—In the insurance example provided in Section 6.6.7, the expectation was that the claims adjusters' productivity would increase over time, as evidenced by a decreased average time to adjudicate a claim. If instead, the productivity "flat lined" or trended downward over several quarters, it is important to find out why this is happening. Research and root cause analysis may reveal one or more reasons for not meeting the business objectives, such as:

- There was a severe spike in the complexity of claims due to natural disasters.
- The solution does not provide sufficient support to assist the adjusters.
- The solution provides sufficient support, but it is awkward to use.
- It takes too much time for a claims adjuster to confirm damages.
- Work is expanding to fill the available time.

6.10.4 Assess Limitations of the Solution and Organization

With analyzed metrics and root causes in hand, many of the same analysis techniques used to understand the problem and organization can be used again to assess the limitations of the solution and the organization.

Example—Referring back to the insurance example,

- A cost-benefit analysis could be formulated to determine whether it is worthwhile to try to improve productivity by adding more claims adjusters during periods of peak claims submittals.
- Further research could be conducted to find out which types of claims take the longest to process and to examine what level of support the solution provides for adjudicating those claims.
- Interviews and usability studies could be conducted to identify awkward aspects of the solution.
- Process flows—perhaps annotated with timings—could be used to identify opportunities for improvement in the claims adjusters' workflow.
- If work expanded to fill the available time, interviews or focus groups could reveal a root cause.

6.10.5 Recommend Approach to Improve Solution Performance

As with other aspects of evaluation, the analysis techniques used to propose the initial solution can also be applied to recommend an approach for improving solution performance. The work to implement the recommended approach needs to be prioritized as part of whatever ongoing planning activities the organization uses.

Example—Continuing with the same insurance example,

- The results of the cost-benefit analysis is input into a recommendation either to add more adjusters during peak claims or to find ways of helping the existing adjusters be more productive (training, additional rule-based processing, etc.).

- The results of the research regarding which types of claims take the longest to process and how well the solution supports adjudicating those claims could lead to recommendations to provide additional training for adjusters or to support additional rule-based processing.
- Awkward aspects of the solution could be redesigned, based on feedback from interviews and usability studies, and then retested.
- Facilitated work sessions could be conducted with decision makers to devise a future-state process flow to improve claims adjuster productivity.
- For work that expands to fill the time available, interviews or focus groups could help formulate a recommendation centered on training or management intervention, without any change to the solution.

6.11 Solution Replacement/Phase out

In general, there are four strategies for solution replacement/phase-out. These strategies apply equally well to automated, manual, and mixed solutions:

- Massive one-time cutover prior to installing the replacement and phasing out whatever existed prior to it.
- Segmented cutover to the replacement prior to phaseout of whatever existed prior to it (e.g., segment by region, by role, opportunistically/on first usage). Segmented cutover implies temporary coexistence of the replacement with whatever is being phased out.
- Time-boxed coexistence of the replacement and what is being phased out, with a final cutover on a specific future date. For this strategy, work conducted with the replacement follows all of the replacement's policies, procedures, and rules, while work conducted with whatever is being phased out continues to operate under its policies, procedures, and rules until the cutover date. This strategy is sometimes used for software projects involving architectural or database changes.
- Permanent coexistence of the old and replacement solutions (with all new business using the replacement solution).

Generally, massive cutovers present more business risk than any kind of cutover that is segmented or time-boxed. Occasionally, however, that risk needs to be accepted.

A cost-benefit analysis can be conducted to determine which approach to use. Whether or not a formal cost-benefit analysis is conducted, some sample questions or research that could be undertaken to determine the most appropriate strategy include:

- *What is the operational impact of having two solutions?* Subjectively, having two solutions may seem like a bad idea from an operational standpoint. However, when there is a great deal of complexity associated with the previous solution, it may be more cost effective to have both solutions coexist with all new business going to the replacement (e.g., when it is difficult to align the previous solution with its replacement and the amount of business conducted using the previous solution is very small).

Example—In the insurance company example, the company created a new consolidated way to handle all life insurance products and decided to retain a small number of individuals to handle claims and questions for a previous product rather than converting that product to the new consolidated solution because: (a) there is only a small amount of life insurance business for the product and it is no longer offered to new customers, (b) the subject matter experts for the previous product are all retired, (c) there is limited documentation to support how to convert the business for this product to the new solution, and (d) it is not possible or practical to exchange policies under that product for policies under another product.

- *Are there any customer-facing or marketing conditions that would require all customers to interact with the new solution at the same time?* Logo changes are an example of a change that would require all business to use the new logo on the same date.
- *Does the replacement solution involve software?* For solutions involving software, cutover usually includes conversion of data from the previous solution. Segmented or time-boxed cutovers are good choices when data conversion is involved.
- *Is it acceptable for some of the customer base to use the previous solution while others use the replacement solution?* Many businesses choose to offer replacement and previous solutions together as part of a segmented or time-boxed replacement strategy. This is especially true where software operating systems and interfaces to software are involved. For example, cellphone users who have the same brand of cellphone may use different versions of its operating system; the new version may be rolled out in segments, where some customers in a specific area may obtain the new version earlier than others; also older phones may not function with the new operating system. The applications offered on these cell phones may have slightly different user interfaces depending on the underlying operating system.

No matter what strategy is selected, care should be taken to develop all of the communication and rollout collateral needed to successfully cutover and adapt to a new solution. These activities include but are not limited to:

- Sufficient communication to the stakeholders who will be directly or indirectly impacted by the new solution;
- Completion of required training materials and delivery of training;
- Completion or update to standard operating procedures (SOPs) and work instructions;
- Purchase, licensing, and installation of any necessary hardware and third-party software needed to support the solution;
- Coordination with other activities within the organization to ensure that implementation occurs at a time when the business can accept the changes, and the rollout is not in conflict with other in-process programs and project work; and
- Coordination to ensure any interruption of business is clearly identified, communicated, and acceptable by customers.

Collaboration Point—When devising communication and rollout activities and collateral, the project managers, training, communication, organizational development, and change management areas should leverage the business analysts' knowledge of the existing solution and the replacement solution as input into their work.

APPENDIX X1
CONTRIBUTORS AND REVIEWERS OF *BUSINESS ANALYSIS FOR PRACTITIONERS: A PRACTICE GUIDE*

The Project Management Institute is grateful to all of these individuals for their support and acknowledges their contributions to the project management profession.

X1.1 Core Committee

The following individuals served as members, were contributors of text or concepts, or served as leaders within the Project Core Committee:

Joy Beatty, CBAP
David P. Bieg
Steven P. Blais, PMP
Susan M. Burk
Elizabeth Larson, PMP, CBAP
Richard Larson, PMP, CBAP
Laura Paton, Chair, PMP, CBAP
Jesse W. Sardina, PMP
Kristin L. Vitello, Standards Project Specialist

X1.2 Content Reviewers

In addition to the members of the Committee, the following individuals provided their review and recommendations on the draft of the practice guide:

Shika Carter, PMP, PgMP, CBAP
Rick Clare, PMP, CBAP
Barbara A. Davis, CBS
Giancarlo Duranti, PMP, CBAP
Nicole Fialkowski
Ellen Gottesdiener, CSM, CPF
Shane Hastie, MIM, CBAP
Peter Lefterov, CBAP

Ginger Levin, PhD, PMP, PgMP
Howard Podeswa
S. Hasnain Rizvi, CBAP, PhD
Jenny Saunders, BSc Hons
Masahiko Soh
Angela M. Wick, PMP, CBAP

X1.3 PMI Standards Program Member Advisory Group (MAG)

The following individuals served as members of the PMI Standards Program Member Advisory Group during development of the *Business Analysis for Practitioners: A Practice Guide:*

Cyndi Snyder Dionisio, MBA, PMP
Larry Goldsmith MBA, PMP
Hagit Landman, PMP, PMI-SP
Yvan Petit, PhD, PfMP
Chris Stevens, PhD
Dave Violette, MPM, PMP
John Zlockie, MBA, PMP, PMI Standards Manager

X1.4 Production Staff

Special mention is due to the following employees of PMI:

Donn Greenberg, Manager, Publications
Roberta Storer, Product Editor
Barbara Walsh, Publications Production Supervisor

APPENDIX X2
BIBLIOGRAPHY

X2.1 Resource Material

The following literature was used in the development of this practice guide:

Alexander, I. F., and Beus-Dukic, L. 2009. *Discovering Requirements: How to Specify Products and* Services. New York: Wiley.

Ambler, S. 2005. *The Elements of UML (2.0) Style.* Cambridge: University Press.

Beatty, J., and Chen, A. 2012. *Visual Models for Software Requirements (Developer Best Practices)*. Redmond, WA: Microsoft Press.

Beatty, J., and Wiegers, K. 2013. *Software Requirements*, 3rd ed. Redmond, WA: Microsoft Press.

Blais, S. 2011. *Business Analysis: Best Practices for Success*. New York: ILL.

Cadle, J. Paul, D., and Turner, P. 2010. *Business Analysis Technique: 72 Essential Tools.* Swindon, UK: British Informatics Society Limited (BISL). Accessed April 8, 2010, http://bcs.org/upload/pdf/business-analysis-techniques_pdf

Carkenord, B. A. 2008. *Seven Steps to Mastering Business Analysis.* Fort Lauderdale, FL: J. Ross Publishing.

Davis, A. M. 2005. *Just Enough Requirements Management: Where Software Development Meets Marketing.* New York: Dorset House Publishing.

Frisendal, T. 2012. *Design Thinking Business Analysis: Business Concept Mapping Applied*. New York: Springer.

Gause, D. C., and Weinberg, G. M. 1990. *Are Your Lights On? How to Figure Out What the Problem REALLY Is*. New York: Dorset House Publishing.

Gottesdiener, E. 2002. *Requirements by Collaboration: Workshops for Defining Needs.* Boston: Addison-Wesley Professional.

Gottesdiener, E., and Gorman, M. 2012. *Discover to Deliver: Agile Product Planning and Analysis*. Sudbury, MA: EBG Consulting, Inc.

Gottesdiener, E. 2005. *The Software Requirements Memory Jogger: A Pocket Guide to Help Software and Business Teams Develop and Manage Requirements*. Salem, NH: Goal/QPC Inc.

Hass, K. B., Wessels, D., and Brennan, K. 2007. *Getting It Right: Business Requirement Analysis Tools and Techniques*. Vienna, VA: Management Concepts.

Hohman, L. 1997. *Journey of the Software Professional: A Sociology of Software Development*. Upper Saddle River, NJ: Prentice Hall Computer.

Jonasson, H. 2012. *Determining Project Requirements: Mastering the BABOK® and the CBAP® Exam.* 2nd ed. Boca Raton, FL: Auerbach Publications.

Kulak, D., and Guiney, E. 2000. *Use Cases: Requirements in Context.* New York: ACM Press.

Larson, E., and Larson, R. *The Practitioners Guide to Requirements Management.* 2nd ed. Minneapolis, MN: Watermark Learning.

Leffingwell, D. and Widrig, D. W. 2003. *Managing Software Requirements: A Use Case Approach.* 2nd ed. Boston: Addison-Wesley Professional.

Lieberman, B. A. 2006. *The Art of Software Modeling*. Boca Raton, FL: Auerbach Publications.

McConnell, S. 1996. *Rapid Development: Taming Wild Software Schedules*. Redmond, WA: Microsoft Press.

Miller, R. E. 2009. *The Quest for Software Requirements*. Milwaukee, WI: Maven Mark Books.

Paul, D., Yeates, D., and Cadle, J., Eds. 2010. *Business Analysis*. 2nd ed. Swindon, UK: British Informatics Society Limited (BISL). Accessed April 8, 2010, http://www.bcs.org/upload/pdf/business-analysis2.pdf

Podeswa, H. 2008. *The Business Analyst's Handbook*. Boston: Cengage Learning PTR.

Rackham, N. 1988. *SPIN Selling*. New York: McGraw-Hill.

Robertson, S. and Robertson, J. 2005. *Mastering the Requirements Process: Getting Requirements Right.* 3rd ed. Boston: Addison-Wesley Professional.

Ross, R. G., and Lam, S. W. 2011. *Building Business Solutions: Business Analysis with Business Rules.* Houston, TX: Business Rules Solutions, LLC

Schneider, G., and Winters, J. *Applying Use Cases, A Practical Guide*. Upper Saddle River, NJ: Pearson Education.

Thayer, R. H., and Dorfman, M. 1990. *System and Software Requirements Engineering*. Washington, DC: IEEE Computer Society Press.

Weinberg, G. B. 1986. *Becoming a Technical Leader*. New York: Dorset House Publishing.

Weinberg, G. M., and Gauss, D. C. 1989. *Exploring Requirements: Quality Before Design*. New York: Dorset House.

Wiegers, K. 2005. *More About Software Requirements: Thorny Issues and Practical Advice (Developer Best Practices)*. Redmond, WA: Microsoft Press, 2005.

X2.2 Suggested Additional Reading Materials

The following are suggested additional reading material:

Alexander I. F., and Beus-Dukic L. 2009. *Discovering Requirements: How to Specify Products and Services*. New York: Wiley.

Bernard M. 2012. *Key Performance Indicators (KPI): The 75 Measures Every Manager Needs to Know*. New York: FT Publishing/Pearson.

Cadle, J. Turner, P., and Paul, D. 2010. *Business Analysis Techniques: 72 Essential Tools For Success Author*. Swindon, UK: British Computer Society.

Cohn, M. 2006. *Agile Estimating and Planning.* Upper Saddle River, NJ: Pearson Education.

Cohn, M. 2009. *Succeeding with Agile: Software Development Using Scrum*. Essex, UK: Addison-Wesley Professional.

Crispin, L., and Gregory, J. 1009. *Agile Testing: A Practical Guide for Testers and Agile Teams.* Upper Saddle River, NJ: Pearson Education.

Davis, A. M. 2005. *Just Enough Requirements Management: Where Software Development Meets Marketing*. New York: Dorset House.

Denne, M., and Cleland-Huang, J. 2004. *Software by Numbers: Low-Risk, High-Return Development*. Upper Saddle River, NJ: Prentice Hall Professional.

Gilb, T. 2005. *Competitive Engineering: A Handbook For Systems Engineering, Requirements Engineering, and Software Engineering Using Planguage.* Oxford: Elsevier Butterworth-Heinemann.

Hubbard, D. 2014. *How To Measure Anything*. 3rd ed. Hoboken, NJ: Wiley.

Kulak, D. and Guiney, E. 2000. *Use Cases: Requirements in Context*. New York: ACM Press.

Paul, D., Yeates, D., and Cadle, J. (Eds.). 2006. *Business Analysis*. Swindon, UK: British Computer Society.

Robertson, S., and Robertson, J. C. 2004. *Requirements-Led Project Management: Discovering David's Slingshot*. Essex, UK: Addison-Wesley Professional.

Rumbaugh, J., Jacobson, I., and Booch, G. 2004. *The Unified Modeling Language Reference Manual*. 2nd ed. Essex, UK: Addison-Wesley Professional.

Wake, B. 2003. *INVEST in Good Stories, and SMART Tasks*. Retrieved from http://xp123.com/articles/invest-in-good-stories-and-smart-tasks/

Young, R, R. 2001. *Effective Requirements Practices*. Essex, UK: Addison-Wesley Professional.

GLOSSARY

Acceptance Criteria. A set of conditions that is required to be met before deliverables are accepted. [Note: In business analysis, acceptance criteria is built to evaluate the product requirements and solution.]

Active Listening. The act of listening completely with all senses so as to pick up all of the information that is being communicated. Active listening entails paraphrasing or reciting back what is heard to ensure accurate understanding of what has been stated.

Activity. A distinct, scheduled portion of work performed during the course of a project.

Adaptive Life Cycle. A project life cycle, also known as change-driven or agile methods, that is intended to facilitate change and requires a high degree of ongoing stakeholder involvement.

Affinity Diagram. A group creativity technique that allows large numbers of ideas to be classified into groups for review and analysis.

Approved Change Request. A change request that has been processed through the integrated change control process and approved.

Architecture. A method to describe an organization by mapping its essential characteristics, such as people, locations, processes, applications, data, and technology.

As-Is Process. A depiction of the current state of a process, representing how a process is currently performed in the organization.

Assumption. A factor that is considered to be true, real, or certain, without proof or demonstration.

Asynchronous Interview. An interview in which the participants are not engaged in the interview at the same time. Asynchronous interviews can be conducted through email or can be prerecorded by the interviewer and provided to the interviewee at a later time.

Backlog. A listing of product requirements and deliverables to be completed, written as stories, and prioritized by the business to manage and organize the project's work.

Baseline. The approved version of a work product that can be changed only through formal change control procedures and is used as a basis for comparison.

Benchmarking. The comparison of actual or planned practices, such as processes and operations, to those of comparable organizations to identify best practices, generate ideas for improvement, and provide a basis for measuring performance.

Brainstorming. A general data gathering and creativity technique that is used to identify risks, ideas, or solutions to issues by using a group of team members or subject matter experts.

Burndown. A visual chart depicting the number of backlog items remaining at any point in time in a project. Burndown charts are used on projects using an adaptive life cycle.

Business Analysis. The set of activities performed to identify business needs; recommend relevant solutions; and elicit, document, and manage requirements.

Business Analysis Approach. A description of how the business analysis process will be conducted for the project or program. The business analysis approach is documented in the business analysis plan.

Business Analysis Center of Excellence. An organizational structure created whereby business analysts are managed centrally or are provided mentorship centrally for the purpose of improving the business analysis discipline across the organization. Also called Center of Business Analysis Practice.

Business Analysis Documentation. The set of business analysis information produced as an output of the business analysis work conducted on a program or project. Such output may be comprised of business analysis deliverables, business analysis work products, or a combination thereof.

Business Analysis Plan. A subplan of the project management plan that defines the business analysis approach, including the tasks that will be performed, the deliverables that will be produced, the roles required to carry out the process, and process decisions regarding how requirement-related decisions will be made; how requirement priorities will be set; how changes to requirements will be proposed, approved, and managed; how requirements will be validated, verified, monitored, and traced; and how business analysis communication will be performed.

Business Analysis Planning. The domain of business analysis that involves planning all of the business analysis activities and reaching the necessary process decisions required for running an effective business analysis process for a program or project.

Business Architecture. A collection of the business functions, organizational structures, locations, and processes of an organization, including documents and depictions of those elements.

Business Case. A documented economic feasibility study used to establish the validity of the benefits of a selected component lacking sufficient definition and used as a basis for the authorization of further project management activities.

Business Need. The impetus for a change in an organization, based on an existing problem or opportunity. The business need provides the rationale for initiating a project or program.

Business Objectives Model. A business analysis model that relates the business problems, business objectives, and top-level features. This model encompasses the justification for a project.

Business Requirements. Requirements that describe the higher-level needs of the organization, such as the business issues or opportunities, and which provide the rationale for why a project is being undertaken.

Business Rule. Constraints about how the organization wants to operate. These constraints are usually enforced by data and/or processes and are under the jurisdiction of the business.

Business Rules Analysis. A process for evaluating, designing, and implementing the rules that govern the organization, its processes, and its data.

Business Rules Catalog. A business analysis model that details all of the business rules and their related attributes.

Business Value. A concept that is unique to each organization and includes tangible and intangible elements. In business analysis, business value is considered the return, in the form of time, money, goods, or intangibles in return for something exchanged.

Capability. The ability to add value or achieve objectives in an organization through a function, process, service, or other proficiency.

Capability Framework. A collection of an organization's capabilities, organized into manageable pieces, similar to a business architecture.

Capability Table. A table that displays the capabilities needed to solve a problem or seize an opportunity. This tool can show the relationship between a situation, its root causes, and the capabilities needed to address the situation.

Cause-and-Effect Diagram. A decomposition technique that helps trace an undesirable effect back to its root cause. See also *fishbone diagram.*

Center of Business Analysis Practice. See *Business Analysis Center of Excellence.*

Change Control. A process whereby modifications to documents, deliverables, or baselines associated with the project are identified, documented, approved, or rejected.

Change Control Board (CCB). A formally chartered group responsible for reviewing, evaluating, approving, delaying, or rejecting changes to the project, and for recording and communicating such decisions.

Change Control Tools. Manual or automated tools used to assist with change and/or configuration management. At a minimum, the tools should support the activities of the change control board (CCB).

Change Management Plan. The plan that defines the process for managing change on the project.

Change Request. A formal proposal to modify any document, deliverable, or baseline.

Charter. See *project charter.*

Closed-Ended Question. A question that calls for a response from a limited list of answer choices. Types of closed-ended questions are forced choice, limited choice, and confirmation.

Communications Management Plan. A component of the project, program, or portfolio management plan that describes how, when, and by whom information about the project will be administered and disseminated.

Configuration Management. A collection of formal documented processes, templates, and documentation used to apply governance to changes to the product, service, result, or subcomponent being developed.

Configuration Management System. A subsystem of the overall project management system. It is a collection of formal documented procedures used to apply technical and administrative direction and surveillance to: identify

and document the functional and physical characteristics of a product, result, service, or component; control any changes to such characteristics; record and report each change and its implementation status; and support the audit of the products, results, or components to verify conformance to requirements. It includes the documentation, tracking systems, and defined approval levels necessary for authorizing and controlling changes.

Constraint. A limiting factor that affects the execution of a project, program, portfolio, or process.

Context Diagram. A visual depiction of the product scope showing a business system (process, equipment, computer system, etc.) and how people and other systems (actors) interact with it.

Contextual Question. A question that can only be answered as it references the subject at hand; namely, the problem domain or the proposed solutions.

Context-Free Question. A question that can be asked in any situation.

Cost-Benefit Analysis. A financial analysis tool used to determine the benefits provided by a project against its costs.

Cost-Effectiveness Feasibility. The high-level economic feasibility of a potential project or program, taking into account both financial benefits and costs.

Data Dictionary. A business analysis model that catalogs the attributes of specific data objects.

Data Flow Diagram. A business analysis model that combines processes, systems, and data to show how data flows through a solution.

Day in the Life Testing (DITL). A semiformal activity, conducted by someone with in-depth business knowledge. The results obtained from DITL testing enable validation or evaluation of whether or not a product or service or solution provides the functionality for a typical day of usage by a role that interacts with the solution.

Decision Table. An analysis model that uses a tabular format to display complex business rules by representing decision points in the upper rows and outcomes in the bottom rows with the purpose of providing all combinations of choices.

Decision Tree. An analysis model that shows business rules associated with complex branching logic. Rules are depicted by modeling the decisions and their outcomes in a tree structure.

Decomposition Diagrams. See *decomposition model.*

Decomposition Model. A model that is used to divide and subdivide a high-level concept into lower-level concepts, for example, dividing the project scope and project deliverables into smaller, more manageable parts for the purpose of analysis. Also known as decomposition diagrams.

Defect Repair. An intentional activity to modify a nonconforming product or product component.

Deliverable. Any unique and verifiable product, result, or capability to perform a service that is required to be produced to complete a process, phase, or project.

Dependency Analysis. A technique that is used to discover dependent relationships.

Display-Action-Response Model. A business analysis model that dissects a screen mockup into its display and behavior requirements at the page element level.

Document Analysis. An elicitation technique that analyzes existing documentation and identifies information relevant to the requirements.

Ecosystem Map. A business analysis model that shows the systems involved in a project and how they interrelate with each other.

Elicitation. See *requirements elicitation.*

Elicitation Plan. An informal device used by a business analyst to prepare for the elicitation work.

Elicitation Session. A session or activity conducted for the purpose of obtaining information from participants. In business analysis, elicitation sessions are conducted in order to obtain the information needed to define the requirements.

Enterprise Architecture. A collection of the business and technology components needed to operate an enterprise. The business architecture is usually a subset of the enterprise architecture and is extended with the applications, information, and supporting technology to form a complete blueprint of an organization.

Entity Relationship Diagram. A business analysis model that shows the business data objects involved in a project and the relationships between those objects, including the cardinality of those relationships.

Estimate. A quantitative assessment of the likely amount or outcome. It is usually applied to project costs, resources, effort, and durations and is usually preceded by a modifier (i.e., preliminary, conceptual, feasibility, order-of-magnitude, definitive). It should always include some indication of accuracy (e.g., ± x %).

Evaluation. See *solution evaluation.*

Expert Judgment. Judgment provided based upon expertise in an application area, Knowledge Area, discipline, industry, etc., as appropriate for the activity being performed. Such expertise may be provided by any group or person with specialized education, knowledge, skill, experience, or training.

Exploratory Testing. An unscripted, free-form validation or evaluation activity conducted by someone with in-depth business or testing knowledge to validate the product and discover product errors.

Facilitated Workshops. An elicitation technique using focused sessions that bring key cross-functional stakeholders together to define product requirements. In business analysis, facilitated workshops use a structured meeting that is led by a skilled, neutral facilitator, in which a carefully selected group of stakeholders collaborate to explore and evaluate product requirements.

Feasibility Analysis. A study that produces a potential recommendation to address business needs. It examines feasibility using one or more of the following variables: operational, technology/system, cost-effectiveness, and timeliness of the potential solution.

Feature. A set of related requirements typically described as a short phrase.

Feature Model. A business analysis model that shows the first, second, and third level of features involved in a project.

Fishbone Diagram. A version of a cause-and-effect diagram that depicts a problem and its root causes in a visual manner. It uses a fish image, listing the problem at the head, with causes and subcauses of the problem represented as bones of the fish. See also *cause-and-effect diagram.*

Five Whys. A technique for conducting root cause analysis suggesting anyone trying to understand a problem needs to ask why it is occurring up to five times to thoroughly understand its causes.

Focus Groups. An elicitation technique that brings together prequalified stakeholders and subject matter experts to learn about their expectations and attitudes about a proposed product, service, or result.

Functional Requirements. Requirements that describe the behaviors of a product.

Gap Analysis. A technique for understanding the gap between current capabilities and needed capabilities. Filling the gap is what comprises a solution recommendation.

Grooming the Backlog. A process used on agile projects where the product team works with the product owner to gain more depth about the user stories in the backlog list. A groomed backlog is an input for sprint planning meetings, which are used to determine which user stories to cover in the next iteration.

Happy Path. See *normal flow.*

High-Fidelity Prototyping. A method of prototyping that creates a functioning representation of the final finished product to the user. High-fidelity prototyping is performed using a programming language or a pseudo language of the product to be demonstrated.

Impact Analysis. A technique for evaluating a change in relation to how it will affect other requirements, the product, the program, and the project.

Iterative Life Cycle. A project life cycle where iterations develop the product through a series of repeated cycles, while increments successively add to the functionality of the product.

Interrelationship Diagram. A special type of cause-and-effect diagram that depicts related causes and effects for a given situation. Interrelationship diagrams help to uncover the most significant causes and effects involved in a situation. See also *cause-and-effect diagram.*

Internal Rate of Return (IRR). The projected annual yield of a project investment, incorporating both initial and ongoing costs into an estimated percentage growth rate a given project is expected to have.

Interviews. A formal or informal approach to elicit information from a group of stakeholders by asking questions and documenting the responses provided by the interviewees.

Ishikawa Diagrams. See *fishbone diagram* and *cause-and-effect diagram.*

Issue. A point or matter in question or in dispute, or a point or matter that is not settled and is under discussion or over which there are opposing views or disagreements.

Iterative Life Cycle. A project life cycle where the project scope is generally determined early in the project life cycle, but time and cost estimates are routinely modified as the project team's understanding of the product increases. Iterations develop the product through a series of repeated cycles, while increments successively add to the functionality of the product.

Job Analysis. A technique used to identify job requirements and the competencies needed to perform effectively in a specific job.

Kanban. An adaptive life cycle in which project work items are pulled from a backlog and started when other project work items are completed. Kanban also establishes work-in-progress limits to constrain the number of work items that can be in-progress at any point in time.

Kanban Board. A tool used within the continuous improvement method of Kanban to visually depict workflow and capacity and assist team members in seeing the work that is planned, in process or completed. The Kanban board is a variation of the original Kanban cards.

Key Performance Indicator (KPI). Metrics usually defined by an organization's executives that are used to evaluate an organization's progress toward meeting its objectives or goals.

Key Stakeholders. A stakeholder who is identified as having a significant stake in the project or program and who holds key responsibilities such as approving requirements or approving changes to product scope.

Lessons Learned. The knowledge gained during a project, which shows how project events were addressed or should be addressed in the future for the purpose of improving future performance.

Low-Fidelity Prototype. A method of prototyping that provides fixed sketches, diagrams, and notes to provide a visual representation of what a screen will look like. Static prototypes do not demonstrate the operation of the system to the user.

Measure. The quantity of some element at a point in time or during a specific time duration, such as the number of work months spent on a project during a specific time period, the number of defects uncovered, or the number of customers responding to a survey stating that they were extremely satisfied.

Metric. A set of quantifiable measures used to evaluate a solution or business.

Model. A visual representation of information, both abstract and specific, which operates under a set of guidelines in order to efficiently arrange and convey a lot of information in an efficient manner.

Modeling Language. A set of models and their syntax. Examples include Requirements Modeling Language (RML), Unified Modeling Language (UML), Business Process Modeling Notation (BPMN), and System Modeling Language (SysML).

Monitoring. The process of collecting project performance data, producing performance measures, and reporting and disseminating performance information.

MoSCoW. A technique used for establishing requirement priorities. In this technique, the participants divide the requirements into four categories of must haves, should haves, could haves, and won't haves.

Multivoting Process. A technique used to facilitate decision making among a group of stakeholders. Participants are provided with a limited number of votes and are asked to apply those votes to a list of possible options. The option with the most votes is determined to be the most favorable option. Multivoting processes can be used to prioritize requirements, determine the most favorable solution, or to identify the most favorable response to a problem.

Narrative. A story. In business analysis, narratives are written when developing personas.

Needs Assessment. The domain of business analysis concerned with understanding business goals and objectives, issues, and opportunities, and recommending proposals to address them.

Negotiation. The process and activities used to resolve disputes through consultations between involved parties.

Net Present Value (NPV). The future value of expected project benefits expressed in the value those benefits have at the time of investment. NPV takes into account current and future costs and benefits, inflation, and the yield that could be obtained through investing in financial instruments as opposed to a project or program.

Nonfunctional Requirements. Requirements that express properties that the product is required to have, including interface, environment, and quality attribute properties.

Normal Flow. Within the context of use case analysis, the normal flow is the set of steps that are followed through the use case scenario when everything goes as planned or expected.

Objective. Something toward which work is to be directed, a strategic position to be attained, a purpose to be achieved, a result to be obtained, a product to be produced, or a service to be performed. In business analysis, objectives are quantifiable outcomes that are desired from a product, result, or service.

Observation. An elicitation technique that provides a direct way of obtaining information about how a process is performed or a product is used by viewing individuals in their own environment performing their jobs or tasks and carrying out processes.

Open-Ended Question. A question that allows the responder to answer in any way desired.

Operational Feasibility. The extent to which a proposed solution meets operational needs and requirements related to a specific situation. It also includes factors such as sustainability, maintainability, supportability, and reliability.

Opportunity. A risk that would have a positive effect on one or more project objectives.

Opportunity Analysis. A study of the major facets of a potential opportunity to determine the viability of successfully launching a new product or service.

Opportunity Cost. The loss of value that could be realized in other actions or alternatives, if the current action is pursued.

Organizational Modeling. A type of modeling that visually depicts the organizational structure and elements of an organization.

Organizational Chart. A model that depicts the reporting structure within an organization or within a part of an organization. In business analysis, organizational charts can be used to help identify stakeholders who are involved in a project and to understand the reporting structures that exist among those identified.

Organizational Goals. Broad-based translations of corporate goals into expressions that are actionable and measurable. Goals are typically longer in scope than objectives.

Organizational Objectives. Accomplishments that an organization wants to achieve to help enable goals. These are specific and tend to be of shorter duration than goals, often one year or less.

Pair-Matching. A step performed when constructing a weighted ranking matrix. It involves taking each option under analysis and comparing it one by one to all the other options listed.

Participant. One who participates in a group activity, such as focus groups or facilitated workshops.

Payback Period (PBP). The time needed to recover a project investment, usually in months or years.

Persona. An archetype user representing a set of similar end users described with their goals, motivations, and representative personal characteristics.

Policy. A structured pattern of actions adopted by an organization such that the organization's policy can be explained as a set of basic principles that govern the organization's conduct.

Phase. See *project phase.*

Predictive Life Cycle. A form of project life cycle in which the project scope, and the time and cost required to deliver that scope, are determined as early in the life cycle as possible.

Problem. An internal or external environment of an organization that is causing detriment to the organization, for example, lost revenue, dissatisfied customers, delays in launching new products, or noncompliance with government regulations.

Problem Domain. The area or context surrounding the problem that is currently under analysis.

Procedure. An established method of accomplishing a consistent performance or result. A procedure typically can be described as the sequence of steps that will be used to execute a process.

Process. A systematic series of activities directed towards causing an end result such that one or more inputs will be acted upon to create one or more outputs.

Process Flow. A business analysis model that visually shows the steps taken in a process by a human user as it interacts with an implementation. A set of steps taken by a system can be shown in a similar model, a system flow.

Process Worker. The stakeholder who physically works with or within the business process that is under analysis or the user who works specifically with a system that is part of the business process. Not all process workers are users.

Product. An artifact that is produced, is quantifiable, and can be either an end item in itself or a component item. Products are also referred to as materials or goods. See also *deliverable.*

Product Backlog. See *backlog.*

Product Scope. The features and functions that characterize a product, service, or result.

Product Stakeholder. A business stakeholder affected by a problem or opportunity, or impacted by or interested in the solution.

Program. A group of related projects, subprograms, and program activities managed in a coordinated way to obtain benefits not available from managing them individually.

Project. A temporary endeavor undertaken to create a unique product, service, or result.

Project Charter. A document issued by the project initiator or sponsor that formally authorizes the existence of a project and provides the project manager with the authority to apply organizational resources to project activities.

Project Life Cycle. The series of phases that a project passes through from its initiation to its closure.

Project Management Plan. The document that describes how the project will be executed, monitored, and controlled.

Project Manager. The person assigned by the performing organization to lead the team that is responsible for achieving the project objectives.

Project Phase. A collection of logically related project activities that culminates in the completion of one or more deliverables.

Project Schedule. An output of a schedule model that presents linked activities with planned dates, durations, milestones, and resources.

Project Scope. The work performed to deliver a product, service, or result with the specified features and functions.

Project Stakeholder Management. Includes the processes required to identify all people or organizations impacted by a project, analyzing stakeholder expectations and impact on the project, and developing appropriate management strategies for effectively engaging stakeholders in project decisions and execution.

Project Team. A set of individuals who support the project manager in performing the work of the project to achieve its objectives.

Prototypes. A method of obtaining early feedback on requirements by providing a working model of the expected product before actually building it.

Questionnaire and Survey. A written set of questions designed to quickly accumulate information from a large number of respondents.

RACI Model. A common type of responsibility assignment matrix that uses responsible, accountable, consult, and inform statuses to define the involvement of stakeholders in project activities.

Regulation. A requirement imposed by a governmental body. These requirements can establish product, process, or service characteristics, including applicable administrative provisions that have government-mandated compliance.

Report Table. A business analysis model that documents in a tabular format all of the requirements necessary to develop a single report.

Requirement. A condition or capability that is required to be present in a product, service, or result to satisfy a contract or other formally imposed specification.

Requirements Attribute. A property of a requirement used to store descriptive information about the requirement, such as last change date, author, source, etc.

Requirements Change Process. The process that defines how changes to requirements will be handled across the project.

Requirements Documentation. A description of how individual requirements meet the business need for the project.

Requirements Analysis. The process of examining, breaking down, and synthesizing information to further understand it, complete it, and improve it.

Requirements Elicitation. The activity of drawing out information from stakeholders and other sources for the purpose of further understanding the needs of the business, to address a problem or opportunity and the stakeholder's preferences and conditions for the solution that will address those needs.

Requirements Elicitation and Analysis. The domain of business analysis concerned with the iterative work to plan, prepare, and conduct the elicitation of information from stakeholders and to analyze, model, and document the results of that work with the objective of defining a set of requirements in sufficient detail to enable the purchase or build of the preferred solution or refinement of processes to achieve the business objective.

Requirements Life Cycle. The flow or life of a requirement throughout a project or program. The requirements life cycle is managed by assigning an attribute or qualifier onto the requirement to depict the requirement state at a specified point in time.

Requirements Management Plan. A component of the project or program management plan that describes how requirements will be analyzed, documented, and managed. See also *business analysis plan.*

Requirement State. An attribute of a requirement that identifies where the requirement falls within the requirements life cycle, for example, in-process, approved, deferred, or rejected.

Requirements Traceability Matrix. A grid that links product requirements from their origin to the deliverables that satisfy them.

Requirements Validation. The process of ensuring that the product satisfies its intended use and anticipated value, ensuring the correct product is delivered.

Requirements Verification. The process of reviewing requirements and models to ensure they meet quality standards. Verification is performed to ensure that requirements are constructed properly and that models conform to the proper use of modeling notation.

Responder. Any participant or person from whom information is gathered by means of elicitation.

Retrospective. A type of meeting in which participants explore their work and outcomes in order to improve both process and product. Retrospectives can occur on a regular basis (e.g., end of iteration or release), at the completion of a milestone, or after a special event (e.g., organizational change, accident).

Return on Investment (ROI). The percent return on an initial project or program investment, calculated by taking the projected average of all net benefits and dividing them by the initial cost.

Risk. An uncertain event or condition that, if it occurs, has a positive or negative effect on one or more project objectives.

Risk Analysis. The process of examining a program, project, or process for risk.

Risk Tolerance. The degree, amount, or volume of risk that an organization or individual will withstand.

Role. A defined function to be performed by a project team member, such as testing, filing, inspecting, or coding.

Rolling Wave Planning. An iterative planning technique in which the work to be accomplished in the near term is planned in detail, while the work in the future is planned at a higher level.

Root Cause Analysis. An analytical technique used to determine the basic underlying reason that causes a variance or a defect or a risk. A root cause may underlie more than one variance or defect or risk.

Scenario. A case of usage of a solution often manifested as a concrete example of a use case or user story or several functional requirements specified in the sequence in which they occur.

Schedule. See *project schedule.*

Scope. The sum of the products, services, and results to be provided as a project. In business analysis, scope is defined as the boundary for the products, services, or results. See also *project scope* and *product scope.*

Scope Baseline. The approved version of a scope statement, work breakdown structure (WBS), and its associated WBS dictionary, that can be changed only through formal change control procedures and is used as a basis for comparison.

Scope Creep. The uncontrolled expansion to a product or project scope without adjustments to time, cost, and resources.

Scope Model. A type of model that identifies the boundaries of the project, program, product, and/or system under analysis. A context diagram is one example of a scope model.

Scrum. A type of adaptive life cycle where a product is built in small incremental portions and each cycle of development builds upon the last version of the product.

Situation. A condition which may be an internal problem or external opportunity that forms the basis of a business need and might result in a project or program to address the condition.

Situation Statement. An objective statement of a problem or opportunity that includes the statement itself, the situation's effect on the organization, and the ultimate impact.

SMART Goals. Goals that are well-written to meet the quality criteria of being specific, measurable, achievable, relevant, and time-bounded.

Software Requirements Specification. A type of requirements documentation that includes the functional and nonfunctional requirements of a software system.

Solution Evaluation. The domain of business analysis concerned with the activities to validate a solution that is about to be or that has already been implemented.

Solution Requirement. A requirement that describes the features, functions, and characteristics of a product, service, or result that will meet the business and stakeholder requirements. Solution requirements are further grouped into functional and nonfunctional requirements.

Sponsor. A person or group who provides resources and support for the project, program, or portfolio and is accountable for enabling success.

Stakeholder. An individual, group, or organization who may affect, be affected by, or perceive itself to be affected by a decision, activity, or outcome of a project, program, or portfolio.

Stakeholder Analysis. A technique of systematically gathering and analyzing quantitative and qualitative information to determine whose interests should be taken into account throughout the project.

Stakeholder Characteristics. The qualities and attributes of a stakeholder, which together determine aspects of how the stakeholder behaves.

Stakeholder Identification. The process of determining the stakeholders impacted by a business problem or opportunity.

Stakeholder Groups. A collection of stakeholders who have similar likes, interests, and stakeholder characteristics. Stakeholder groups are used by project managers and business analysts to manage large groups of stakeholders.

Stakeholder Map. A technique used to visually analyze stakeholders and their relationship to each other and to the problem or opportunity under analysis.

Stakeholder Register. A project document including the identification, assessment, and classification of project stakeholders.

Stakeholder Requirement. A requirement that describes the need of a stakeholder or stakeholder group.

State Diagram. A business analysis model that visually shows how an object moves between different states. This model helps to show the life cycle of an object in a solution.

State Table. A business analysis model that shows all of the possible states of an object and all of the valid transitions. This model helps to enumerate all possible states and possible transitions.

Subject Matter Expert (SME). A person who is considered an expert in a particular subject area. In business analysis, SMEs are often involved in providing the requirements for their area of expertise.

Surveys. See *questionnaires and surveys.*

SWOT Analysis. Analysis of strengths, weaknesses, opportunities, and threats of an organization, project, or option.

Synchronous Interview. An interview conducted where the interviewer and interviewee are involved in the interview at the same time. Synchronous meetings can occur face to face, over the phone, or through web conferencing, etc.; the two parties are not required to be in person with one another, but both need to be active in the interview at the same time.

System Feasibility. See *technology feasibility.*

System Interface Table. A business analysis model that documents the requirements for the connections between each interfacing system involved in a project, including how they are connected and what information flows between them.

Technique. A defined systematic procedure employed by a human resource to perform an activity that produces a product or result or delivers a service, and that may employ one or more tools.

Technology Feasibility. An analysis to determine the extent to which a technology exists in an organization to support a potential solution and if not present, how feasible it would be to acquire and operate the needed technology.

Templates. A partially completed document in a predefined format that provides a defined structure for collecting, organizing, and presenting information and data.

Threat. A risk that would have a negative effect on one or more project objectives.

Time Feasibility. An analysis to determine how well a proposed solution can be delivered to meet the organization's needed time frame.

To-Be Process. A proposed revision to an existing process that can provide an improvement for an organization over how activities are currently performed, or a revision to a new process when adding products or services.

Traceability. Traceability provides the ability to track product requirements from their origin to the deliverables that satisfy them.

Traceability and Monitoring. The domain of business analysis concerned with building and maintaining the traceability matrix to manage requirements and product scope, baselining the product requirements, assessing impacts of proposed requirement changes, and managing the required updates to the requirements and other business analysis deliverables once proposed changes are approved.

Traceability Matrix. See *requirements traceability matrix.*

Transition Requirements. Requirements that are the temporary capabilities, such as data conversion and training requirements, needed to transition from the current as-is state to the future state.

Use Case. An analysis model that describes a flow of actor-system interactions and boundaries for those interactions, including trigger, initiating and participating actors, and preconditions and post conditions.

Use Case Diagram. A business analysis model that shows all of the in-scope use cases for a project and which actors have a part in those use cases.

User Class. A group of stakeholders who are users of a software system, product, or service and are grouped together due to the similarity in their requirements and use of the product.

User Experience Analyst. Also referred to as user interface analysts; individuals who are responsible for studying user behavior, preferences, and constraints in order to identify user interface and usability requirements for software applications and other products.

User Interface Flow. A business analysis model that shows the specific pages or screens of an application and how a user can navigate between them.

User Story. A one or two sentence description, written from the viewpoint of the actor, describing what function is needed. A user story usually takes the form of "as an <actor>, I want to <function>, so that I can <benefit>."

Validation. The assurance that a product, service, or system meets the needs of the customer and other identified stakeholders. It often involves acceptance and suitability with external customers. Contrast with *verification.*

Verification. The evaluation of whether or not a product, service, or system complies with a regulation, requirement, specification, or imposed condition. It is often an internal process. Contrast with *validation.*

Velocity. A measure of a team's productivity rate at which the deliverables are produced, validated, and accepted within a predefined interval. Velocity is a capacity planning approach frequently used to forecast future project work.

Version Control. The process of maintaining a history of changes on software or documentation.

Version Control System (VCS). A system that is used to track the history of revisions, often but not always related to software.

Weighted Criteria. A technique used to help support objective decision making. It uses a weighted ranking matrix to compare alternatives and their weighted scores in order to evaluate decision options. See also *weighted ranking matrix.*

Weighted Ranking Matrix. A table used in decision making that combines pair matching of all alternatives with weighted criteria to add objectivity when formulating a decision or recommendation. Each alternative is compared with every other alternative on the basis of weighted criteria, and the resulting scores are added together to determine the preferred choice.

Work Breakdown Structure (WBS). A hierarchical decomposition of the total scope of work to be carried out by the project team to accomplish the project objectives and create the required deliverables.

Work Product. An output produced as a result of some completion of work that is required for a short-term purpose and not required to be monitored and maintained on an ongoing basis.

INDEX

