

NYS Business Analysis Guidebook

Contents

Articles

Business Analysis Guidebook	1
Introduction	7
The Business Analyst Role	7
Keys and Barriers to Business Analyst Success	8
Maturity Models for Business Analysis and Self-Assessment	8
Stakeholder Analysis	12
Building a Business Case	13
Documenting and Managing Requirements	14
Requirement Gathering Tools	24
Business Analysis Within a Typical System Development Life Cycle	32
Data Modeling and Data Documentation	48
System Acceptance, Test Planning and Strategy	50
Implementation and Training	54
User Experience	54
Business Process Improvement Models	55
Root Cause Analysis	56
Strategic Analysis and Planning	62
LEAN	68
Facilitation and Elicitation Techniques	71
Prioritization Techniques	77
BA Software Tools	79
Communication Skills	80
Building Relationships and Trust	83
Analytical Thinking and Problem Solving	83
Creativity	87
Procurement, Including RFPs, RFIs and IFBs	87
Business Intelligence and Performance Management	89
Requirements Templates	89
Glossary	89
Resources	93
Noted Contributors	93
Contributors	94
List of Figures	94
Licenses	94

References

Article Sources and Contributors	106
Image Sources, Licenses and Contributors	107

Article Licenses

License	108
---------	-----

Business Analysis Guidebook

Guidebook Introduction

This Business Analysis Guidebook is designed to facilitate a consistent approach in the use of the tools and techniques contained within the Business Analyst profession. The primary goal is to provide a simple "how to" guide for new and non-Business Analysts for gathering (eliciting) and documenting business requirements--whether they are at the process, project or enterprise level. The material linked below was initially authored by a group of Business Analyst professionals within NYS Government, and it is hoped that the list of contributors (included in later chapters) will grow. We are currently weaving our content together--so please pardon our duplication and inconsistent formatting. It is our intent to have this material be licensed under Creative Commons Attribution-Share Alike 3.0 Unported License and the GNU Free Documentation License, both to be included at the end of this document.

Contents

1. Introduction
 1. Foreword
 2. About this Book
 2. The Business Analyst Role
 1. What is a Business Analyst
 2. Project Manager versus Business Analyst and When You are Both
 3. Keys and Barriers to Business Analyst Success
 4. Maturity Models for Business Analysis and Self-Assessment
 1. Analyst Maturity
 1. Business Analyst Aptitude Questionnaire
 2. Organizational Maturity
 1. Business Analysis Maturity Model
 2. Capability Maturity Model Integration
 3. Business Analysis Center of Excellence
 3. Further Reading
 4. References
 5. Stakeholder Analysis
 6. Building a Business Case
 1. Why Create a Business Case?
 2. Why a Sponsor is Critical for the Project
 3. Parts of a Business Case
 1. Why the Project is Necessary
 2. Options Considered
 3. Benefits of Doing the Project
 4. Costs
 5. Timeframe
 6. Cost Benefit Analysis
 7. Risks
 8. Evaluate the Proposal
 7. Documenting and Managing Requirements
-

1. Requirements Development
 1. Gather/Elicit Requirements
 1. Learn How the Organization Operates
 2. Learn How to Speak the Organizations Language
 3. PPlan to Capture Requirements
 2. Requirements Sources
 1. Business Rules vs. Project Requirements
 2. Business Case/ITIR
 3. Constraints (Policies, Standards, Legal)
 4. Existing Applications
 5. Business Users/Stakeholders
 3. Analyze Requirements
 1. Categorization
 2. Dependencies
 3. Impact and Feasibility Analysis
 2. Requirements Management
 1. Capture Requirements
 2. Validate Requirements
 3. Manage Requirements
 4. Trace Requirements
 5. Communications
 1. Appropriate
 2. Artifacts
 3. Delivery Method
 4. Communication Changes
 8. Requirement Gathering Tools
 1. What are the Tools?
 1. Process Models
 2. Use Cases, User Stories, Business Scenarios
 3. Graphical Illustration
 4. Prototypes and Mockups
 5. Data Dictionary
 6. Glossary
 7. Business Forms
 2. Why Use These Tools?
 3. How to Use These Tools
 1. Focus for Requirements Facilitation and Elicitation
 2. Other Uses for Tools
 4. How to Create Some Diagrams
 1. Visio - Templates, BPMN Stencil Template
 2. Excel - Formatting, Page Setup
 3. Microsoft Word
 5. General tips
 9. Business Analysis Within a Typical System Development Life Cycle
 1. Introduction
-

-
2. System Development Lifecycle Methodologies (Workflow Patterns)
 3. System Development Lifecycle
 1. System Initiation Phase
 2. System Requirements Analysis Phase
 1. Prepare for System Requirements Analysis
 2. Determine Functional and Non-Functional Requirements
 1. Functional Requirements
 2. Non-Functional Requirements
 3. Non-Functional Requirements Deliverables
 4. Requirements and SDLC Methodologies
 3. Define Process Models
 4. Define Logical Data Model
 5. Reconcile Functional and Non-Functional Requirements with Models
 3. System Design
 4. System Construction
 5. System Acceptance
 6. System Implementation
 4. Software Quality Assurance (SQA)
 5. Project Roles and Responsibilities
 6. SDLC at a Glance
 7. References
 10. Data Modeling and Data Documentation
 1. Business Data
 1. Data Governance and Security
 2. Business Ownership
 3. Eliminate Redundancy
 2. Data Flows
 1. Where Does the Data Come From?
 2. What Processes Transform the Data?
 3. Where Does the Data Go?
 3. Data Documentation
 1. Process Diagrams - (Maps and Data Flow)
 2. Data Definitions
 3. Object Role Model
 4. Entity Relationship Model
 5. Datamart and Data Warehouses
 11. System Acceptance, Test Planning and Strategy
 1. Testing Roles and Responsibilities
 2. Test Planning - The Test Strategy Document
 3. Developing the Test Plan
 4. Priority Testing
 5. Regression Testing
 6. Positive and Negative Testing
 7. Volume and Load Testing
 8. Environments
 9. Defect Management and Tracking
-

10. User Acceptance Testing (UAT)
 11. System Acceptance
 12. Implementation and Training
 1. Implementation
 1. Implementation Plan
 2. Communication and Coordination
 3. Migration Scheduling (Pre- and Post-Migration Testing)
 2. Training
 1. Planning - Scope, Resources, Webinars, Site Visits, Facilities
 2. User Documentation - Manuals, Training Documentation, Release Notes
 13. User Experience
 1. What is Usability, User Experience (UX)
 1. Content - Plain Language - Reading Level - Jargon
 2. Where to Start - Statistics, Sensitivity, Easy Fixes
 3. Engineering
 1. Design, Navigation, User Research, Above the Fold, Layout Considerations
 4. Testing
 1. UAT, Test Labs, Soliciting Customers, Techniques
 14. Business Process Improvement Models
 1. What is Business Process Improvement (BPI)
 2. What is Business Process Machine Notation (BPMN)
 3. Life Cycle
 4. How Does Business Analysis Fit in?
 5. Best Practices
 6. Performance Metrics and Key Performance Indicators (KPIs)
 15. Root Cause Analysis
 1. What is Root Cause Analysis and Why is it Important?
 2. When is it Used?
 3. Questions to Consider
 4. Sources of Problems
 5. Methods
 1. Fishbone Diagram
 2. Ask Why 5 Times
 3. Check Sheets and Pareto Chart
 4. Interrelationship Diagram
 5. Rapid Problem Resolution
 6. Summary
 7. References
 16. Strategic Analysis and Planning
 1. Strategic vs. Tactical vs. Operational
 2. Strategic Planning Components
 1. Mission and Vision
 2. Values and Behaviors
 3. SWOT Analysis and Environmental Scans
 4. Stakeholder Analysis
-

5. Goals and Objectives
 6. Metrics
 7. Strategic Projects
 3. Enterprise Analysis - Strategic Alignment of Business and IT Goals
 4. Session Planning Guidelines
 5. Techniques to Engage Staff
 6. Facilitating the Generation/Acceptance of the Plan
 1. Writing and Communicating the Plan
 2. Implementing the Plan
 17. LEAN
 1. LEAN
 2. Six Sigma
 3. LEAN for Government
 4. Value Stream Mapping
 5. For More Information
 18. Facilitation and Elicitation Techniques
 1. Introduction
 2. Facilitation Basics
 1. Planning
 2. Conducting
 3. Follow Up
 3. Facilitation Techniques
 4. Troubleshooting and Dealing With Difficult Behaviors
 5. Requirements Elicitation Techniques
 1. Vision Development Techniques
 2. Definition Techniques
 3. Analysis Techniques
 4. Other Techniques
 19. Prioritization Techniques
 1. Prioritization and When to Use?
 2. Prioritization Techniques
 1. Strategic Alignment
 2. High-Medium-Low or Small-Medium-Large
 3. Weighted
 4. Ranking
 5. Criteria
 6. Matrix
 7. Impact Analysis
 3. Validating
 20. BA Software Tools
 1. Introduction
 2. Business Process Management (BPM) and Diagramming Tools
 1. Microsoft Visio
 2. Open Text - ProVision
 3. Global 360
-

-
4. BizAgi
 5. Tibco
 6. iGrafx
 7. IBM Tools
 8. Adobe Live Process Management
 9. Progress Savvion Modeler
 3. Requirements Management Tools
 1. Rational Requisite Pro
 2. Microsoft Word and Excel
 3. PPM Studio
 4. Blueprint
 5. Accompa
 4. Drawing Tools
 1. Microsoft Visio
 2. SmartDraw
 21. Communication Skills
 1. Introduction
 2. Communication Types
 3. Communication Skills
 4. Acquiring Communication Skills
 22. Building Relationships and Trust
 1. Why Trust is Important in Organizations
 1. What is Trust and what it is Not
 2. What Creates a Culture of Mistrust?
 3. How to Build Trust over Time
 23. Analytical Thinking and Problem Solving
 1. Underlying Competency (Assess, Understand, Make Judgments on a Solution)
 1. Creative Thinking
 2. Decision Making
 3. Problem Solving
 24. Creativity
 1. Creativity and Its Role in Business Analysis
 1. Importance of Creativity
 2. Balancing Left Brain and Right Brain Thinking
 2. Creativity Techniques
 1. Brainstorming and Brainwriting
 2. Checklists
 3. What-If Scenarios
 4. Reformulating the Problem
 5. Role Playing
 6. Provocation Techniques
 7. Mapping
 8. Reverse Engineering
 25. Procurement, Including RFPs, RFIs and IFBs
 1. Procurement Guidelines
-

1. Types of Procurements - Including Pros and Cons
2. Key Steps and Timelines
3. Guidelines for Writing a Statement of Work (SOW)
4. Considerations When Evaluating Proposals
26. Business Intelligence and Performance Management
27. Requirements Templates
28. Glossary
29. Resources
30. Noted Contributors
31. Contributors
32. List of Figures
33. Licenses
 1. Creative Commons Attribution-Share Alike 3.0 Unported License
 2. GNU Free Documentation License

Introduction

The Business Analyst Role

What is a Business Analyst?

According to the International Institute of Business Analysis (IIBA), a business analyst works as a liaison among stakeholders in order to elicit, analyze, communicate, and validate requirements for changes to business processes, policies, and information systems. A variety of roles are covered by this definition and many titles are used to describe those roles which add to the confusion. Some examples of different types include:

- Business analysts with very strong business skills and understanding of the business domain whose key role is to analyze business processes, procedures, etc. in order to identify problems and determine solutions. These analysts are more involved in what the IIBA defines as enterprise analysis and are likely to be involved prior to the initiation of an information technology (IT) project.
- IT Business Analyst who is focused on requirements elicitation and analysis, and solving problems using information technology solutions. This analyst serves as a bridge between business and IT and generally begins work after a project has been initiated. This analyst specifies “what” the system must do.
- Systems Analyst is an IT business analyst who is more focused on system design and the technical aspects of the solution. This analyst takes the requirements and creates functional specifications regarding “how” a system will do the “what.”
- Many other titles are used including the Business Systems Analyst which has been described as a combination of the IT Business Analyst and the Systems Analyst.

The most important element is the business focus; ensuring business needs are understood and communicated so that the final solution meets the business needs. Solutions may be IT related, non-IT related, or could be process improvement. The business analyst is responsible for eliciting the actual needs of stakeholders, not simply their expressed desires and often play a central role in aligning the needs of business units with the capabilities delivered by information technology.

Project Manager Versus Business Analyst and When You Are Both

The best way to succeed on any type of project is to have a strong, experienced Project Manager (PM) and a strong, experienced Business Analyst (BA). Working together from the beginning, they set the stage for success by accurately planning and clearly defining the expected outcomes. Each role provides specialized capabilities and is responsible for a different set of tasks. The PM keeps an eye on the management of the project, ensuring the project delivers on time, on budget and with the full scope of the requirements met. The BA focuses on management of the business need, business requirements, and expected business benefits with an eye on the final product the project delivers. It is the project manager who owns the Business Solution Life Cycle, and the business analyst who owns the Systems Requirements Life Cycle, from understanding the business need to ensuring that the delivered solution meets the need and adds value to the bottom line. Excellent PMs and BAs will work together to make the most of each other's strengths.

On some projects one person is required to act as both the PM and the BA. This is often the case on smaller projects. For the individual, the challenge is to be aware of the conflicting focus and to try to act in one role at a time. For larger projects playing both roles puts the project at risk for either rushing requirements elicitation and analysis tasks and missing important requirements or spending too much time working on requirements and jeopardizing the project schedule.

Keys and Barriers to Business Analyst Success

Keys and Barriers to Business Analyst Success

Maturity Models for Business Analysis and Self-Assessment

Maturity Models for Business Analysis and Self-Assessment Models

Analyst Maturity

Today Business Analysts may come from within organizations or from consulting firms. Often those from within the organization have strong backgrounds in either the business or its IT department. Regardless of background, there are four skill sets that any Business Analyst will strive to improve:

- Understanding of the business, its culture, and its domain (e.g., government)
- Understanding of the principles of information technology, the IT within the organization, and the trends in the IT field
- Business analysis techniques and tools
- Personal qualities and behavioral skills

The first is extremely important to business analysis. Much time is given to understanding the organizational structure, its mission, resources, output, and the framework in which it operates (non-profit, government, etc.)

Secondly, a strong understanding of IT principles is needed as most business solutions involve IT systems. These principles include how information technology works (computers, networks, internet, cloud, etc.), system development processes (Agile, DSDM, etc.), off-the-shelf products, and trends in technological opportunities for business / government. Many community colleges and technical schools offer introductory or overview coursework.

Thirdly, BA techniques are well documented and can be found in many books, magazines and webinars. Techniques include investigation, stakeholder assessment, elicitation, business process and IT systems analysis, requirements gathering, data modeling, facilitation, presentation, project management, change management, and strategic analysis. Behavioral skills are the essential keys for a successful business analyst. These skills can be grouped as inter-personal and analytical skills.

- The Business Analyst must have good communication skills in order to obtain data and then present a proposed course of action, all the while reducing the anxiety of change. These skills include good listening, empathy, elicitation, common language, and the ability to adjust the language to the audience. Good communication skills build relationships, positively influence others, develop cohesive team work, and effect confidence. [reference # 23 Communication Skills]
- Analytical thinking characteristics are problem-solving, attention to detail, big-picture views, procedural orientation, organization in collecting and analyzing data, and both conceptual and factual thinking. Critical thinkers will dig deeper and deeper and sift through conflicting information to find the true situation and the real business need. With experience, the Business Analyst will even be able to pre-assess the degree of analysis needed. [reference # 26 Analytical thinking]

With these basic capabilities and a love of learning, the Analyst will continually improve techniques. The following suitability questionnaire – adapted by the New York State Police from Barbara K. Carkenord, MBA, CBAP - will assist the new Analyst or anyone thinking about entering the field. For those currently in the role it may also reveal root causes for any frustration (or success). The more questions you agree with, the greater is your potential in succeeding in a Business Analyst role. Taking this non-scientific questionnaire may reveal a need for a better sense of your own qualities and personalities. There are self-assessment tools available for this discovery:

- Birkman Method® assesses interpersonal style, interests, underlying motivations, and reactions to stress
- DISC® assesses behavioral style (Dominance, Influence, Steadiness, Conscientiousness) and describes the intensity of each
- Myers-Briggs (MBTI®) assesses personality dichotomies including extroversion vs introversion, sensing vs intuition, thinking vs feeling, and judgment vs perception.

Agency BA Aptitude Questionnaire

Do I have an aptitude for performing business analysis? Take the following aptitude test; the questions were adapted for government use from original material developed by Barbara K. Carkenord, MBA, CBAP.

QUESTION	USUALLY	SOMETIMES	RARELY
1. I regularly organize information such as my finances or recipes.			
2. I am a planner; I usually go shopping with a list and I have at least a general plan for a vacation.			
3. I prepared documents that are organized, concise, and clear.			
4. I am good at drawing diagrams such as flow charts and floor plans.			
5. I am able to break down and simplify a complex topic.			
6. I often have a To Do list of tasks to accomplish.			
7. I enjoy learning new ideas and procedures.			
8. I enjoy puzzles and problem solving.			
9. I enjoy getting into the details.			
10. I can step back and look at the big picture.			
11. I enjoy working with people.			
12. I can motivate myself to get things done.			

13. I can prioritize tasks.			
14. I look for improvement opportunities such as constructive criticism.			
15. I can remain calm when others are over-stimulated.			
16. I am comfortable dealing with conflict at work.			
17. I am patient with others as they try to understand concepts.			
18. I can politely tell people when they are straying from the main point of a story.			
19. I like to know the goal of conversations and planning sessions.			
20. I am good at negotiating solutions among people.			
21. I prefer NOT to manage or supervise people.			
22. I do not need to be the center of attention but can take the lead when needed.			
23. I am good at leading a meeting and keeping everyone on topic and schedule.			
24. I am comfortable making presentations in front of groups.			
25. I enjoy working on long and complex projects.			
26. I learn the culture and the attitudes of people to determine the frame of reference in which I work.			
27. I learn the jargon and vernacular of the culture in which I will work.			
28. Before I start a task, I think about the process, timing and the goal.			
29. I tend to point out the similarities in conversations before the differences.			
30. I get more satisfaction from the process than the closure.			
31. I take time to thoughtfully respond to an email question or opinion rather than reacting.			
32. People seldom have to ask for clarification after receiving my email.			
33. When giving a formal presentation, people usually indicate they understand my message.			
34. I enjoy helping people learn new things.			
35. I create positive relationships with people.			
36. I can easily change my language to better reach other people.			
37. I believe I am a good trouble-shooter.			
38. I am comfortable as a team player and a team leader.			
39. People enjoy working with me and help when I ask.			
40. People often ask me for help.			
Total number in each column:			

Typically, the higher the counts in the column labeled "Usually", the more you are suitable to a Business Analysis role or position.

The beginner with an aptitude for business analysis may first take on small parts of small projects such as documenting current business activities and then building business requirements. After a few years of experience and mentoring, the Analyst may take on full BA responsibilities for small or medium projects and assist in larger ones. With 5 to 10 years experience, the Analyst may take the lead in large scale, mission-critical projects. An Analyst with over 10 years of experience may be called upon to participate in enterprise strategic planning or manage a center of excellence.

Organizational Maturity

The path to maturity for an organization follows that of the individual Analyst. Business analysis has evolved from the software development. Historically software development has had long timelines during which many executives were not always patient. Even in government, the program managers might not wait for the IT department to fill the technology need and often purchased off-the-shelf products and training. IT departments began adding a business systems analysis function so that the deliverables were clearly defined and timelines were more accurately projected. The programs themselves began adding business analysts to the projects so that the business needs could be better understood and conveyed properly.

As the business analysis function broadens within organizations a common progression can be observed. The “Business Analysis Maturity Model” (BAMM™) was developed by Assist Knowledge Development Limited in the United Kingdom. <http://www.assistkd.com/knowledge-hub/business-analysis-maturity-model/bamm/>. This model identifies three stages of improvement that have a direct correlation to the complexity of the work and the authority (degree of influence) given to the Analyst:

System Improvement	At this lowest level, requirements are defined for an IT system, the scope of the work is delineated, and the Analyst's authority is limited to the project itself.
Process Improvement	At this level, process analysis is added by investigating the processes that give rise to the requirements; the Analyst's authority is expanded to the underlying processes for which the IT system is a solution.
Business Improvement	At this level, the scope of the project and the Analyst's authority are high and analysis involves enterprise strategies and senior level management.

A more widely known maturity model is the “Capability Maturity Model Integration” (CMMI®). <http://resources.sei.cmu.edu/library/asset-view.cfm?assetID=18556>. This model was developed as a quality standard of organizational process improvement and software development by the Software Engineering Institute of Carnegie Mellon University. These five levels of the CMMI® can be observed within each of the three stages of BAMM™. The CMMI® model is as follows:

Level 1	Initial	processes and structures are not controlled and are mostly ad-hoc and reactive
Level 2	Managed	processes are characterized for projects but are often reactive
Level 3	Defined	processes are characterized for projects and are proactive, i.e., processes are built on the organization's standards
Level 4	Quantitatively Managed	processes are controlled and measured
Level 5	Optimizing	processes work smoothly and focus moves to improvement

A highly developed organization will have a progressive “Business Analysis Center of Excellence” (BACoE). A Center of Excellence of any content or focus has as its primary goals:

- To develop competencies in its resources and provide tools and structure,
- To strive for quality analysis and develop standards to measure quality, and
- To participate in enterprise strategic planning.

An agency with an active BACoE would enjoy a high project success rate.

Kathleen Hass has developed a four-stage model to describe the growth of the BA function within organizations which includes a BACoE. See http://www.loriusllc.com/images/BA_Practice_Maturity_White_Paper_2010.pdf

Awareness	the organization is simply aware of business analysis but does not demonstrate an understanding of the value it adds; Analysts participate in a BACoP
Framework	the organization is fully supportive of the BA function and establishes a BACoE using BABOK® standards. The CoE assigns roles and responsibilities for the BA function. It establishes the tools and processes for managing all aspects of business requirements. The CoE also develops training, measurements, and improvement management processes.
Alignment	the organization vests accountability in its BACoE by missioning it to manage resources (employees, contractors, vendors) and to develop and implement tools for business alignment including enterprise analysis, portfolio management, strategic alignment of projects, solutions assessment, and benefit measurement
Optimization	the organization integrates the BACoE with other CoEs such as Project Management and Quality Assurance so that business opportunities are optimized using technology solutions. At this level, the BACoE develops, trains, and implements such tools as customer relations management, organizational readiness management, and organizational change management

Centers of Excellence emerged in the 1990s in the project management area. A BACoE is often born when a few analysts align to share techniques and processes. This group - or any Analyst - may benefit from joining a BA Community of Practice (BACoP) in the area. A BACoP brings business analysts from many agencies and businesses together to share on a larger scale.

There are other maturity models available today. Each may address specific angles and use focused language but they all display a pattern of growth from ad-hoc to a fully structured entity integrated into the enterprise as a power center. It should not be difficult to place your organization on the spectrum; the challenge is to move it to the next stage of maturity.

Stakeholder Analysis

Placeholder for Stakeholder Analysis

Thank you Elaine!

Building a Business Case

Building a Business Case OUTLINE

Developing a Solid Business Case

Identify and Quantify a Problem

- Define the business need
- Define the project parameters
- Determine if there is a viable solution
- What are the merits of the problem
- Would the project be consistent with the agency's mission and strategic plan
- Present the problem and proposed solution in a context that allows management to use consistent criteria to compare all project proposals

Parts of the Business Case

Reason Why the Project is Necessary

- Define the project parameters
- Describe the As-Is
- State the problem
- Tell a story, make it interesting, build a persuasive case,
 - i. clear
 - ii. concise
 - iii. factual – include all facts
 - iv. objective
 - v. minimal use of jargon and conjecture
- Create a picture of the end state
 - i. Show the value, financial justification
- High-level view of the proposed project that will allow all affected units in the organization to understand the purpose and be knowledgeable about the project
- Possible future state: How will the business function with and without the proposed solution
- Interview as many people as necessary to understand the big picture and desired outcome

Options

- Develop a potential solution
- Described other possible solutions, and why they were rejected

Benefits of Doing the Project

- Link issues to solutions and benefits
- Improves customer satisfaction?

Explain how that will happen, what the effects will be on the business

Costs

- Money
- Personnel

Cost-Benefit Analysis

- Value for money,
- return on investment
- show cost/benefit over a defined period of time
- will the proposed solution impact other units or processes (+ or -)

Risks

- Identify risks to project success

Evaluate the Proposal

- Why is the project necessary
 - What are the top benefits
 - Why is your proposed project the best way to attain the desired results
- Any previous projects to solve similar problems?

Documenting and Managing Requirements

In this Section of the Guidebook, readers should gain an understanding of the activities that are associated with the documentation and management of project and application requirements. These activities support an organized methodology for performing Business Analysis throughout a project life cycle and enable an analyst to position an organization to manage application projects and downstream maintenance in support of organizational strategies and work processes efficiently and effectively. Whether an analyst is involved in a project or initiative from the ground level or 'Initiation' through enhancement changes or only involved in a part of the development life cycle, information contained in this section should provide guidance for the activities involved in the requirements management process.

Documenting and Managing Requirements

Organizations spend a lot of money on key projects to ensure continued viability in a rapidly changing world. They invest in projects to solve a business problem, take advantage of an opportunity or implement a strategic solution that furthers business goals. Capturing and managing the right requirements ensures avoidance of costly missteps and is key to delivering successful projects with measureable value.

Requirements Development

Requirements development includes activities related to gathering and analyzing the requirements.

Gather/Elicit Requirements

When gathering requirements, Business Analysts need to capture information within the context of the organization and in support of operational needs to satisfying stakeholder goals.

Learn How the Organization Operates

The process of gathering and eliciting requirements begins with an understanding of how an organization operates. Most projects involve a small subset of the overall organization. Understanding the high-level process flow of a business highlights the ‘big picture’ and serves as a foundation for planning the requirements gathering and management efforts. In public sector operations, the primary mission is to serve the public. Whether that involves ensuring the health and safety of the public, enforcement of laws and regulations, providing support services, or some other mission, identifying the high-level purposes that drive the project environment ensures that the project outcomes are aligned with the organization’s goals.

The construction of a context to identify the project’s organizational impact involves several iterations. Each iteration explores different organizational aspects until those operations directly affected by the scope of the project are isolated and there is a clear understanding of the scope of the expected deliverable(s). Communicating with business users and eliciting the sources of this information is a key step in building your plan for requirements management.

Learn How to Speak the Organization’s Language

Terminology: The study of terms and their use
Semantics: The study of <i>meaning</i>

Familiarizing yourself with the business terminology is a fundamental component of establishing an informed understanding of a project domain. The semantics and terminology of the business ‘language’ bridges gaps between the business, technical and external stakeholders. Using a common terminology/semantics to communicate information about requirements ensures that the deliverable will fulfill the expectations of all stakeholders.

Your role as a successful liaison depends in part upon expanding your vocabulary to meet the project needs. The internet and an organization’s intranet can be resources for gaining the additional vocabulary needed to assist in communicating clearly with all project stakeholders.

It is always useful to capture important terms and acronyms used within the requirements in a requirements glossary, especially where resources are shared and/or outsourced. Such a glossary may be held at the organizational level – if so, it can be a valuable resource for you to help in understanding the language of the business and as an aid to writing clear and concise requirements.

Plan to Capture Requirements

Requirements are useful throughout the SDLC and into the future. During the execution of a project, your requirements will provide a source for analysis in performing the impact analysis for change requests, design changes and other typical project experiences. Capturing requirements in an organized way enables you to perform fully informed analysis throughout the project life cycle and beyond, into the application maintenance phase.

When you have an existing application and need to incorporate an enhancement to that application’s functionality, your requirements can be re-used to determine every functional area of the application where proposed changes may impact the existing application. Planning for current project and ongoing maintenance deliverables requires the BA to form a plan for organizing the requirements.[1]

Choosing appropriate attributes for your stored requirements will facilitate your analysis downstream during project execution. Attributes, like Source, Status, Functional Area, etc., are pieces of information about your requirements and can be used to filter requirements so that analysis is focused on only relevant information. Is the project complex enough to warrant this effort? To what level? Make decisions concerning how you will organize and manage

requirements before the project is in full 'Requirements Gathering' mode.

A document repository for project documentation can be used to ensure that a project team has one source of authority for project artifacts like process models, use cases, key presentations, base-lined requirements, project and work plans, defects and change logs, or other documents used in the project execution. This type of tool enables project team members and business stakeholders to reference source material for requirements on demand, promoting a diversified review of the resulting project work.

Requirements Sources

There are many sources of requirements. BAs examine the business operational information and communicate with stakeholders and other interested parties to gather requirements. Many will be uncovered during the organizational analysis phase of gathering requirements. Even project documentation includes additional requirements, stated at a high level.

Business Rules

There is a common misconception that business rules are requirements. While there may be a one-to-one correspondence between the two, there is a distinction between them. Requirements describe what is needed to implement the business rules. Business rules themselves describe how the business must operate. The following table provides a comparison.

Business Rule	Requirement
Customers may reserve an appointment by scheduling a date and time to meet with a representative of the department.	<ul style="list-style-type: none"> • The public facing web site will display a phone number customers may call to reserve an appointment. • The public facing web site will allow customers to navigate to an appointment reservation request form. • The reservation request form will include the customer name, address, phone and e-mail contact information and requested appointment date and time. • The customer must not be able to choose an appointment date and time on the reservation request form that is already booked. • Customers may submit their request directly from the form interface. • Submitted reservation request forms will be scheduled on the department calendar.
Failure to pay the fee by the due date will result in a .02% penalty charge to the customer.	<ul style="list-style-type: none"> • The application will automatically determine when a customer fee is overdue based on the current date and the receivable due date. • When a customer fee is overdue, the application will calculate and update the penalty amount on the customer receivable record. • The penalty calculated for overdue fees will be equal to .0002 * the overdue customer receivable balance + previously accrued fees.

Business Rules (BABOK v2.0)[2]
• Define or constrain some aspect of the business
• Apply across processes and procedures (system agnostic)
• Intended to assert business structure or control/influence the behavior of the business
• Rules are atomic – that is, they cannot be broken down

Many organizations actively manage their business rules. There are also many organizations where the rules are embedded in the policies and procedures used to guide operations. Many procedures are not captured in any document, but reside within the staff knowledge-base. Regardless of which environment you are working in, the business rules affecting your project should be identified to ensure that your project deliverables support overall operations.

For those organizations that effectively manage their business rules, understanding them is fairly straightforward. Where an organization does not formally manage the rules, you will need to perform analysis to identify the rules. Understanding applicable policies and procedures, and accepted IT and Security standards, prepares you to ensure

that the requirements expressed by your users comply with business operations that must be supported by the project deliverable.

Business Case/ITIR

The Business Case, Project Charter, Information Technology Investment Request (ITIR) or other documentation created during project initiation clearly outlines the project deliverable expectations and justification. This document will generally provide high-level business requirements that you will use to guide the development of functional and non-functional requirements. It may even include some of the functional and non-functional requirements themselves. This is the starting point for user requirement discovery, providing a context for discussions and communications.

Constraints

Identifying constraints that are applicable to the project ensures that the deliverables are feasible. No software application can be constructed or implemented without an understanding of the technical environment where it will reside. No business application can be constructed or implemented without an understanding of the legal environment where it will reside. The solution must support the technical IT policies and security standards, and the law and regulations applicable to the business and/or business process. Constraints affecting the solution are important to identify non-functional requirements and to use when communicating with project stakeholders during the development of the functional requirements.

Existing Applications

When a project goal is to replace or enhance an existing application, that application can provide many of the requirements that will be applicable to the solution. This 'As-Is' scenario is a gold mine for determining necessary functionality and project requirements. If requirements were managed for the old application, they can be re-used, with the appropriate suitable modifications, for the current project. Maybe the project deliverable addresses a brand new need that no existing application supports. If this is the case, the project solution may need to incorporate business rules and functional requirements for multiple existing applications and business processes. This type of 'As-Is' scenario analysis is typically applicable to system integration projects.

Business Users/Stakeholders

While investigations into the Organizational structure, business processes, rules and project artifacts provide a foundational understanding of the business problem or opportunity that the project was created to address, it is an analysis that is constrained to a single perspective – that of the analyst. Business users and other project stakeholders provide verification that the requirements are correct and they also provide previously unidentified requirements that are not necessarily found in organizational documents, process analysis or other method. Gathering these types of requirements requires communications, involving interviews, surveys and Requirements Gathering sessions. Reconciling and incorporating the multiple perspectives of users and stakeholders is an important BA activity.

Analyze Requirements

As requirements are gathered they should be analyzed within the context of the other requirements, the overall project and the organization. This activity will identify requirements that are not valid, highlight missing or incomplete requirements and provide the necessary information to complete requirement definitions.

Categorization

As requirements are captured, category information should be recorded as requirements attributes. Categories assigned to requirements assist in filtering requirements for analysis purposes. The context of the project will determine how and to what extent requirements are organized into categories. A primary goal of categorizing requirements is to facilitate analysis. Some organizations may have standard requirements categories to use while other use project-specific categorization. Categorization supports quick assessment of the impact of proposed changes and supports traceability for downstream work efforts, such as design and testing, helping to guide activities.

Use categories that make sense to the project where possible. For example, a development effort that involves multiple, integrated applications may want to include a category for the individual application the requirement applies to. Then, you can easily examine only those requirements associated with the single application when analyzing work efforts, costs, etc. For a small, self-contained application project, this category would not add any value. Regardless of what categories are used, they should provide an organized way to facilitate analysis, minimize development rework and ensure that the objectives for the project are delivered.

Dependencies

Dependencies that exist between individual requirements help to guide prioritization and highlight possible risks to the project's success. For example, if you have one requirement that the customer address will be captured and another requirement stating that the system will assign work to staff based on where the customer lives, then the 'work assignment' requirement depends on the customer address requirement. From this dependency analysis, you now know that the customer address **MUST/SHALL** be captured before the work assignment requirement will be met. It also provides incentive to refine the address requirement to include all relevant address elements.

Requirements may also have dependencies on other projects or the existing infrastructure. For example, if your project will interface with the deliverable for another project, you may have requirements that are dependent on the output of the other project. If that project fails, your requirement may need to be modified or invalidated.

Impact and Feasibility Analysis

Organizational Impact
Extends beyond project focus and may include:
• infrastructure and technology strategies
• other applications
• work process changes

Implemented requirements generate changes to the organization, technical architecture, security, business processes and/or groups of people (both internal and external). The project team will be able to mitigate risks, set expectations and prevent unexpected consequences by understanding how the project will affect these organizational variables. Impact analysis includes identifying the impact if a requirement is not implemented, as well as the impact if it is included in the project deliverables. Use this information to set priorities and provide change management guidance.

During the requirements elicitation and facilitation process, you may identify requirements that certainly are possible, but they are not practical. A simple checklist can be used to focus attention on those requirements that

affect areas of concern. The level of impact (None, Low, Medium, High) can be estimated and decisions regarding the feasibility of accepting a requirement can be guided by this analysis.

Where the project plan incorporates a staggered implementation of the deliverable or the project will be executed using an Agile development methodology, impact analysis may be needed to determine which requirements should be implemented for development cycle. Factors to consider when performing impact and/or feasibility analysis include:

- Associated costs,
- Complexity of implementing the requirement,
- Skill levels of the technical development team and the users who will use the application and
- The organization's operational ability to support the completed project deliverables.

These factors can provide requirement attribute definitions (i.e., Is the cost associated with implementing the requirement 'High', 'Medium' or 'Low?'), contributing to categorization of requirements.

The project manager and business sponsors should carefully review 'High' impact changes to verify their validity, cost, feasibility, effect on operations and any other factors discovered during the impact analysis. This assessment should prevent unexpected results during and following the execution of a project.

Requirements Management

Managing requirements provides the foundation for project analysis, design, development, quality assurance and ongoing maintenance activities. The primary tasks associated with requirements management involve capturing requirements for re-use, validating requirements, managing the change associated with requirements and ensuring traceability of the deliverable to the organizational goals and needs.

Capture Requirements

Capture the requirements to manage them throughout the project lifecycle and for re-use into the future. The level of a requirement may identify the appropriate attributes that should be captured. For example, associated costs may not be a realistic attribute for a detailed system requirements, but support prioritization for a high-level Business requirement. For each requirement, identify and record information (attributes) about the requirement that is relevant to the project. The List of Commonly Used Requirements Attributes at right lists some common attributes that are often included in a requirements repository and include the appropriate requirement level where it makes sense to capture the attribute. Please note that these attributes represent general guidelines and should be adjusted for the project scope.

Word processing documents are often good vehicles for communicating requirements across the project's stakeholder groups. But using a document-based approach to managing requirements introduces issues that can be avoided by storing requirements, and requirement attributes, in a spreadsheet, simple database or other requirements management tool. These issues include:

- Documents are difficult to keep current
- Changes to requirements become hard to communicate
- Information needed to manage requirements is difficult to store and use
- It is difficult to trace the requirements to other project artifacts [3]

Attribute	Description	High-Level	Mid-Level	Detail (Testable)
Unique ID	A unique identifier for the requirement	X	X	X
Date Created	Date the requirement was first captured	X	X	X
Source/Ownership	Name of the person, organizational division or change request where the requirement came from	X	X	X
Functionality	Functional capability or constraint (Non-functional category)	X	X	X
Requirement Type	Functional/Business, Non-functional/Technical, Transition, Stakeholder	X	X	X
Requirement	Concise description of requirements expressed in business terminology to ensure common understanding	X	X	X
Status	Reflects the current state of the requirement. Examples: Proposed, In Review, Approved, Denied, Verified, Implemented	X	X	X
Priority	Indicates the importance of the requirement (Critical, High, Medium, Low)	X	X	X
Rationale/Origin	Reflects backward (justifying source) traceability and justifies the requirement	X	X	X
Difficulty/Complexity	Qualitative measure of how difficult the requirement is to implement	X	X	X
Refrain	Indicates how mature the requirement is - can it be worked on?	X	X	X
Test Case ID	Reflects forward traceability to validation test case used during the Acceptance phase of design	X	X	X
Assigned To	The person responsible for completing the requirement	X	X	X
Risk	Assigned based on the expected associated level of effort and/or complexity/difficulty	X	X	X
Impact	Business or technical effect of implementing this requirement	X	X	X
Due Date	Target date for the deliverable associated with the requirement	X	X	X
Stakeholders	Identified business staff responsible for approving changes that impact the requirement	X	X	X
Approved By	Person who approved this requirement	X	X	X
Approval Date	Date the requirement was approved	X	X	X
Release Number/Build	Application release number targeted for the implementation of this requirement	X	X	X
Dependencies	Identification of other system elements and/or requirements that may be impacted by changes to this requirement	X	X	X
Comments	Text field available for analyst, approver, etc. notes related to this requirement	X	X	X
Verification	Specification of how this requirement will be tested (e.g., demo, analysis, inspection, walkthrough, QA test script/test cases)	X	X	X
Subsystem(s)	Applications/systems where this requirement will be implemented	X	X	X
Actual Release	Identifies the Actual, rather than the Planned Version (Release Number/Build) where the requirement was implemented	X	X	X
Version Number	Used to track different versions of the requirement	X	X	X
Cost	Quantified cost attributable to this requirement	X	X	X
Defect	COV of any associated defects	X	X	X
Obsolete	Indicates that the requirement has been removed/is no longer active	X	X	X
Benefit	Need or goal that this requirement supports	X	X	X
ROI/ROE	Associated return on investment/effort - that may be expected when this requirement is implemented	X	X	X

List of commonly used Requirements Attributes

What should be the extent and level of requirements captured? This is generally determined on a case-by-case basis. It depends on various factors, including the nature of the business, the stakeholders who are involved, the complexity/scope of the deliverables and the project management methodology. A project that involves implementing an off-the-shelf application, for example, would generally not document all requirements as extensively as a project that is charged with integrating the information from multiple databases. Each project must determine the appropriate level of requirements definition based on the project deliverables.

The level at which requirements are captured should be driven by the project complexity and environment. At a minimum, the high-level business requirements must be captured. These requirements clearly and concisely state the needs and goals of the solution, providing a common understanding across stakeholder groups. They represent the foundation for validation of the solution during the quality assurance activities performed before release. They provide the source for 'backward traceability' of the detailed requirements that are implemented.

During the requirements definition phase of a project, the Functional and Non-Functional, or Business and Technical, requirements are identified and refined. These types of requirements may be captured at a very detailed level (e.g., UI, User, Stakeholder, etc.) or at a higher level which only encapsulates the 'why' and 'what' of the solution. Each lower-level requirement should be directly associated with the high-level business requirement that it supports. This ensures complete validation of the solution.

Transitional requirements[4] may also be needed to cover the cut-over from the existing processes/applications to the new solution that the project will deliver.

Functional/Business	define the behavior and capabilities of an application, the information that the application will manage and the required inputs/outputs. These requirements must take into account how a business operates, the improvements and changes necessary to support the project goals and needs, and any existing constraints that must be supported.
Non-Functional/Technical	must support, and be supported by the environmental conditions under which the application must remain effective. They include the qualities that the application must have. They encompass criteria related to performance, scalability, security, usability, system availability and the underlying information architecture.
Transitional	exist only during the transition from the 'As-Is' state to the 'To-Be' state of the project deliverables and are not identified until the system is designed. They include data transformation, user training and other transition needs that must be supported during the implementation of the solution.

Verify Requirements

After the initial discovery, there is generally time involved to gather associated information and finalize the requirements. Each requirement should have a complete set of attributes captured and express the business need, or 'what?', that will be supported. The actual requirement text should be written clearly and concisely so that all project reviewers and approvers will share a common understanding of each requirement.

The process that results in requirements approval involves all project team members, business experts and technical oversight. Requirements are distributed to those designated as responsible for review. Reviewers will accept requirements as written, dispute requirement information, elaborate on the rationale or dependencies, and generally provide feedback to correct or complete requirements. It is important for requirement reviewers to understand their role in this process. Clear direction for reviewers should be provided, including criteria for evaluation and sign-off procedures for validation.

Once the requirements are reviewed and adjusted for any corrections or missing information, they must be approved by designated operational and/or technical authorities. The approval process involves a review of the operational and technical accuracy of the requirements and considers the feasibility to implement. An approved requirement will provide a 'baseline' for purposes of the requirements management process, setting user and other stakeholder expectations for the project deliverable. For IT projects where the deliverable will be accomplished in phases or sprints, approval for the requirements that will be supported in later phases of a project does not have to be completed before initial development work can begin.

During the period of review and approval, a process should be used to resolve conflicts and/or issues. This process

will provide a framework to support forward movement on the project. Each project may have a unique process for resolving problems, depending on the business, culture and team members involved. Regardless of the process that is established, it should be clearly communicated to all project team members. And then, it should be followed. Consistency in the resolution of problems will enhance the ability to execute the project and meet objectives.

Manage Change

As the project progresses, there will be further changes and/or refinements that affect the approved requirements and/or create new requirements. Changes may be generated by the identification of new required functionalities, errors or defects in the current implementation and high priority requests to enhance existing functionalities. Each potential change should be analyzed to determine associated feasibility, costs and benefits. An analysis of the impact of making the change should identify any associated changes to project schedules, costs or other existing requirements.

Change requests are often captured separately from the project requirements. Identifying the requirements that will be affected by the change is critical to maintaining correct and current requirements for the project. Change requests should include details to justify enhancements or identify the necessity for the associated work. In effect, each request must include a mini-business case to justify the change. The Description of Attributes Collected for Managing Changes table at right lists some suggested change request attributes. As with the captured requirements attributes, change attributes should be determined to incorporate applicable project scope and needs.

Managing the change involves capturing the change, gaining stakeholder approval, modifying existing project plans, process or other diagrams/models, and requirements to include the change, executing the change, performing QA on the change and, finally, implementing it. This process will impact existing requirements and create new project requirements. The changes to your baseline requirements should be identified using the unique change request ID number as the requirement (or modified requirement) 'Source'.

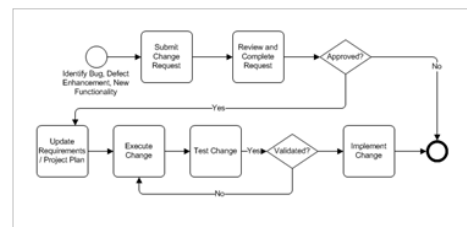
When a requirement is 'Approved', it becomes a baseline requirement. This 'Approved' status remains in effect until and unless it is changed. For example, a change request is approved for implementation that will modify our existing 'Approved' requirement. Now the existing baseline requirement will be replaced using an updated version of the requirement that incorporates the approved change. This updated version of the requirement is the new baseline, the currently 'Approved' requirement. The previously approved requirement is retired. Other project deliverables that are dependent on the requirement, including quality assurance tests and user documentation, will also need to be updated for the approved revision.

A simplified process that might be used for managing changes is illustrated in the Change Management Process diagram at right. Information about the change, including scope, feasibility, costs, benefits and impacts to existing requirements and technical architecture, will be needed for the approver review. Specific designees will be responsible for approving changes.

Once a change request has been approved, the change will be incorporated into the current project work schedule, budget and requirements repository. For new requirements, attributes will be captured in the requirements repository. For changes to existing approved requirements, the original requirement(s) should be replaced with the altered requirement(s). The original requirement(s) should be marked with an inactive status and retained for historical analysis purposes.

Request Attribute	Description
Unique ID	A unique identifier for the change request.
Date Created	Date that the change request was first captured.
Requestor	Name of the person or organizational division where the request originated.
Requester Contact Info	Phone, e-mail contact information for Requestor.
Change Description	Statement of what needs to be changed and why.
Analysis	Analysis of what application functionalities/features and what existing requirements will be affected by the implementation of the requested change.
Scope	Identify the system modules or components that will be changed from their current state.
Impact	List any impacts to business processes, user experience, other integrated applications, etc. to be considered for implementing this change.
Benefits/Costs	Includes a mini-business case for the change; identify any cost-related risks.
Status	Reflects the current state of the change. Examples: proposed, approved, deleted/rejected, analysis, development, testing, implemented.
Assessment of Effort	Estimated resource time needed to complete the change and implement; identify any resource-related risks, scheduling conflicts.
Approver	Identified business staff responsible for approving changes that impact the requirement.
Approve Date	Date that approval was given for developing the change.
Assigned To	The person responsible for completing/monitoring the change.
Due Date	Target date for the implementation of the deliverable associated with the change.
QA Methodology	Specification of how this change will be tested (e.g., demo, analysis, inspection, walkthrough, QA test scripts, etc.).
QA Approval Date	Date that the change passed QA inspection/testing.
Release Number	Application release number targeted for the implementation of this change.
Dependencies	Identification of system elements or requirements that may be impacted by this change.
Notes	Text field available for analyst, reviewer, approver, etc. notes related to this change.
Subsystem(s)	Applications, systems where this requirement will be implemented.

Description of Attributes Collected For Managing Changes

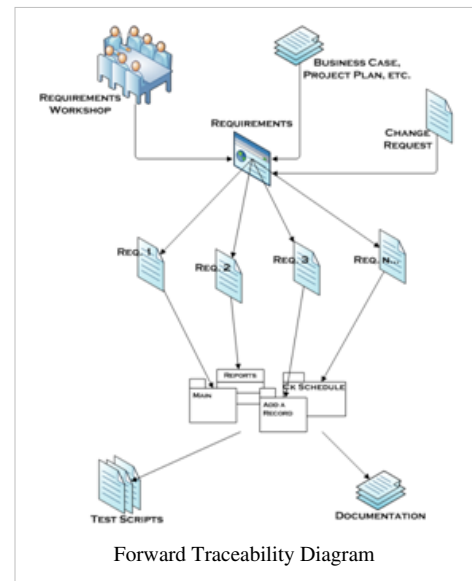


Trace Requirements

Requirements are the foundation for ensuring that a project deliverable supports the needs that justified the project. Each captured requirement should include an attribute that provides 'backward' traceability to the source of the requirement.

Once requirements are approved, other project artifacts will be generated. There is a correspondence between the requirements and project work. For IT projects, the requirements will be allocated to different areas of the designed deliverable. For non-IT projects, the requirements also must be satisfied by project deliverables. Quality testing and user documentation and training also are derived from the requirements. Traceability, from the original project documents through the project life cycle to the implemented solution, should be supported in the captured requirements and other project artifacts.

The diagram at right shows the 'forward' traceability from requirement sources, through the design of the deliverable, test scripts for the project's QA process and end user documentation. The flow can be reversed to review the 'backward' traceability path. Using a methodology that supports this tracing capability provides the means to ensure that the project deliverables support all the identified project needs and goals.



Traceability is usually presented using a matrix that correlates any two base-lined project artifacts. It is especially useful when determining the impact a change may generate. The Matrix at right provides an example of a Requirements Traceability matrix that might be used when assessing the impact of a requirement change to the quality testing for new development work.

In this example, a change to the functionality related to the third 'Form'-related functional requirement has the potential to affect the 23 existing functional validation tests for the Creation, Editing and Editing/Copying form tasks.

As with the capturing of requirements, the degree to which requirements are traced should be determined by the circumstances of the project. Maintaining a very high level of traceability may be overkill when the associated effort delivers a low business value. But failure to maintain adequate traceability can easily impact the ability to remain within the cost and schedule constraints of the project.

Requirements Test Count: Test by Task

Task	Functional Requirement ID															
	FR1	FR2	FR3	FR4	FR5	FR6	FR7	FR8	FR9	FR10	FR11	FR12	FR13	FR14	FR15	FR16
Task 1: Create New Record	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Task 2: Associate Providers																
Task 3: Associate Providers																
Task 4: Associate Providers																
Task 5: Associate Providers																
Task 6: Associate Providers																
Task 7: Associate Providers																
Task 8: Associate Providers																
Task 9: Associate Providers																
Task 10: Associate Providers																
Task 11: Associate Providers																
Task 12: Associate Providers																
Task 13: Associate Providers																
Task 14: Associate Providers																
Task 15: Associate Providers																
Task 16: Associate Providers																
Task 17: Associate Providers																
Task 18: Associate Providers																
Task 19: Associate Providers																
Task 20: Associate Providers																
Task 21: Associate Providers																
Task 22: Associate Providers																
Task 23: Associate Providers																
Task 24: Associate Providers																
Task 25: Associate Providers																
Task 26: Associate Providers																
Task 27: Associate Providers																
Task 28: Associate Providers																
Task 29: Associate Providers																
Task 30: Associate Providers																
Task 31: Associate Providers																
Task 32: Associate Providers																
Task 33: Associate Providers																
Task 34: Associate Providers																
Task 35: Associate Providers																
Task 36: Associate Providers																
Task 37: Associate Providers																
Task 38: Associate Providers																
Task 39: Associate Providers																
Task 40: Associate Providers																
Task 41: Associate Providers																
Task 42: Associate Providers																
Task 43: Associate Providers																
Task 44: Associate Providers																
Task 45: Associate Providers																
Task 46: Associate Providers																
Task 47: Associate Providers																
Task 48: Associate Providers																
Task 49: Associate Providers																
Task 50: Associate Providers																
Task 51: Associate Providers																
Task 52: Associate Providers																
Task 53: Associate Providers																
Task 54: Associate Providers																
Task 55: Associate Providers																
Task 56: Associate Providers																
Task 57: Associate Providers																
Task 58: Associate Providers																
Task 59: Associate Providers																
Task 60: Associate Providers																
Task 61: Associate Providers																
Task 62: Associate Providers																
Task 63: Associate Providers																
Task 64: Associate Providers																
Task 65: Associate Providers																
Task 66: Associate Providers																
Task 67: Associate Providers																
Task 68: Associate Providers																
Task 69: Associate Providers																
Task 70: Associate Providers																
Task 71: Associate Providers																
Task 72: Associate Providers																
Task 73: Associate Providers																
Task 74: Associate Providers																
Task 75: Associate Providers																
Task 76: Associate Providers																
Task 77: Associate Providers																
Task 78: Associate Providers																
Task 79: Associate Providers																
Task 80: Associate Providers																
Task 81: Associate Providers																
Task 82: Associate Providers																
Task 83: Associate Providers																
Task 84: Associate Providers																
Task 85: Associate Providers																
Task 86: Associate Providers																
Task 87: Associate Providers																
Task 88: Associate Providers																
Task 89: Associate Providers																
Task 90: Associate Providers																
Task 91: Associate Providers																
Task 92: Associate Providers																
Task 93: Associate Providers																
Task 94: Associate Providers																
Task 95: Associate Providers																
Task 96: Associate Providers																
Task 97: Associate Providers																
Task 98: Associate Providers																
Task 99: Associate Providers																
Task 100: Associate Providers																

Matrix showing tasks against functional requirements in order to facilitate traceability

Communications

The communications necessary during the requirements gathering and management activities will be determined by the project purpose, stakeholders and environment. For smaller projects, a requirements communication plan may be deemed as not necessary; for larger projects, a formal plan may be an essential ingredient to ensure that all team and management staff are coordinated in their efforts and that all project objectives are being met.

Regardless of the level of formality of the project, the goals associated with requirements communications center around the concept that all stakeholders should share a common understanding of the requirements.

Appropriate to the Recipient

Communications should always be presented at a level that is appropriate to the recipient of the information. The level of communication required for individual stakeholders should be matched to the role the person plays in the execution of the project.

For example, the Executives may receive an Executive Summary Requirements Document that presents the requirements at a very high level. SMEs and other stakeholders may review the mid-level functional requirements that describe what inputs, outputs, behavior and capabilities are needed. Detailed functional and system requirements are needed by the development and quality assurance staff.

Using these various levels when communicating about requirements will prevent recipients from receiving too much information, or not enough, for their purposes.

Requirements Artifacts

Where formal requirements artifacts are produced, these should follow standard formatting and include consistent content to accomplish the work task that they support. Standards regarding format and content may be adopted from a number of Standards-setting bodies or they may be developed internally within an organization. Consistency in the presentation of requirements information will facilitate the shared understanding across all stakeholder groups.

Figure r lists some common artifacts that are used for requirements management. The actual requirements artifacts that are used during the execution of a project may include these artifacts, and/or others, as appropriate for the project.

EDIT: Put in diagram re: Attribute Artifacts

Requirement Information Delivery

Communications that occur during the Requirements Gathering phase of a project often take the form of informal conversations and e-mail messages between the project team and the business customers. Some projects may have a shared repository where documents can be reviewed simultaneously by multiple people, and may include electronic approvals; other projects may depend on stakeholders receiving e-mailed documents to be printed and filed in project binders. Regardless of the delivery method, it is a primary Business Analyst function to ensure that all project stakeholders have the appropriate requirements information to perform their roles within the project.

Communicate Requirement Changes

Changes to base-lined requirements must be communicated to the project stakeholders. Much of this communication is done during the course of the approval process, but for those not involved in that process, notifications, presentations or other methods of communication should be used so that all are aware of the change and its impact on the project work. Changes may need to be applied to upstream and/or downstream project deliverables. For example, a change to a law may require modifications to high-level project documents, requirements document, design plan, testing plan and/or other formal artifacts. When a change that includes such revisions is made, the people that use or rely on the artifact should be notified that the artifact has changed and why the change happened. The notification should include the updated revision of the affected artifact documents.

Application of material provided in this Section:

The scope of analysis work may include BA participation for the entire performance of a project or be limited to only a piece of the total project work. Documenting and managing requirements provides a framework to accomplish the necessary activities throughout the project lifecycle. The benefits of documenting requirements in an organized fashion include the ability to perform such tasks as impact analysis for changes, providing complete scope information for creating user guidance, determining quality assurance tests to validate the business goals and support for an organized approach to managing project team communications. Investing the effort to capture requirements at the time project requirements are being defined pays off even for future development projects involving the project deliverables.

References

- ^ 1. Software-Quality-Assurance.org, CMMi - Requirements Management (REQM), retrieved 6/12/2012 from <http://www.software-quality-assurance.org/cmmi-requirements-management.html>
- ^ 2. Business Analysis Body of Knowledge (BABOK Guide), v2.0, Section 9.4.2
- ^ 3. Weigers, Karl E., Automating Requirements Management, retrieved 6/5/2012 from http://www.processimpact.com/articles/rm_tools.html.
- ^ 4. Business Analysis Body of Knowledge (BABOK Guide), v2.0, Section 7.4.

References

- [1] http://en.wikibooks.org/wiki/Business_Analysis_Guidebook/Documenting_and_Managing_Requirements#endnote_CMMI._2012
- [2] http://en.wikibooks.org/wiki/Business_Analysis_Guidebook/Documenting_and_Managing_Requirements#endnote_IIBA._9.4.2_2010
- [3] http://en.wikibooks.org/wiki/Business_Analysis_Guidebook/Documenting_and_Managing_Requirements#endnote_Weigers._2012
- [4] http://en.wikibooks.org/wiki/Business_Analysis_Guidebook/Documenting_and_Managing_Requirements#endnote_IIBA._7.4_2010

Requirement Gathering Tools

Requirements Gathering Tools

A variety of tools are used to assist in the requirements gathering process. Each type of tool provides alternative means to illustrate, explain and specify exactly what must be delivered to meet the business goals. They simplify the understanding of requirements by application of the truism ‘a picture is worth a thousand words’. They encompass process documentation, graphical illustrations and detailed specifications to help in eliciting requirements, communicate proposals and decisions, provide detailed specifications and identify missing or incomplete requirements.

This section uses examples for the many requirements that might be associated with the goal of acquiring a new car. The customer must decide what car to buy, the car dealer needs to be able to provide the car to the customer and the legal compliance requirements applied to ensure that public goals for safety and information are met during the transaction. Each of these players has different needs and constraints that must be satisfied to accomplish the delivery of a new car to the customer. Examples provided below will illustrate the tools that might be used to capture the requirements for this scenario.

What are the Tools?

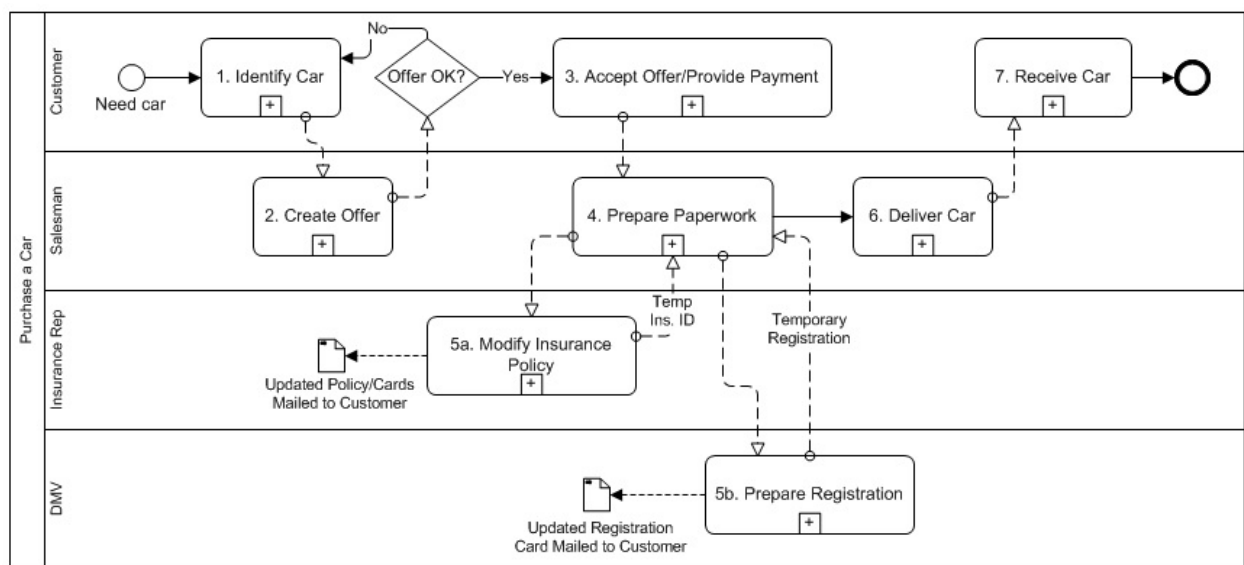
Tools included in the examples below illustrate many options that are available to a business analyst for use in gathering requirements. During the course of a project, one or more of these tools may be appropriate to use for gathering and/or clarifying/validating the requirements. There are additional modeling tools available which are not covered here, such as data-/task-/work-flow models, application or infrastructure diagrams and activity diagrams. Tools included here are readily understandable by all project stakeholders.

Process Models

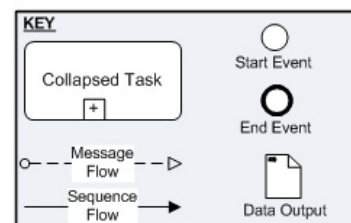
Process models provide a visual representation of a series of tasks, activities or actions directed toward specific goals. They are useful for illustrating a range of complexity, from providing a very high-level view of the overall process (Figure n) to capturing detailed activities for a small piece of the overall process (Figure n+1). These figures are constructed below using the Object Management Group (OMG) Business Process Model and Notation (BPMN) specification, version 2.0 (<http://www.omg.org/spec/BPMN/>).

High-level diagrams identify the context for the low-level process models. Inputs and outputs serve as placeholders for data requirements, indicating important information. Process delays and operational issues can be included in this type of diagram to assist in the identification and assessment activities involved in Business Process Improvement (BPI) initiatives.

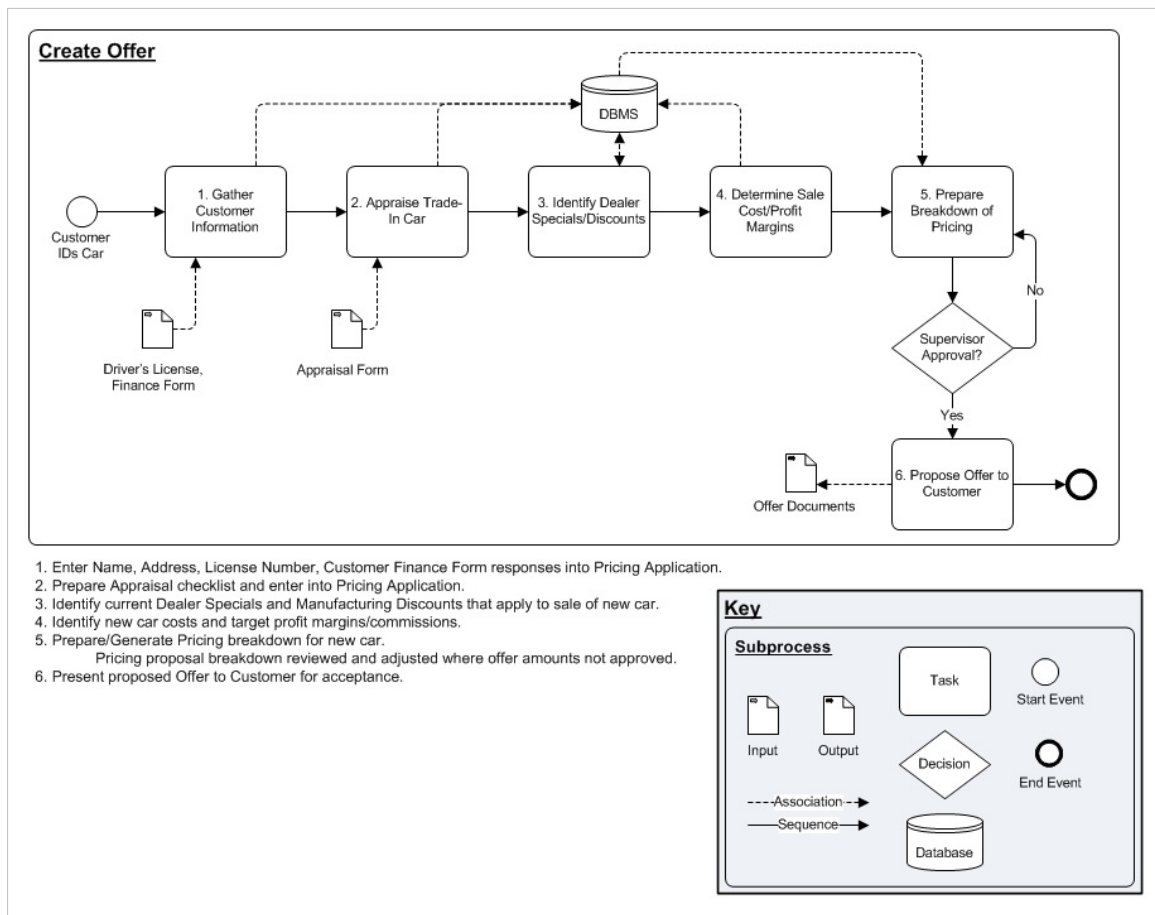
High-level Process Model: Simple process outlines what happens when a car is purchased.



1. Customer identifies car to for potential purchase based on applicable factors (e.g., price, expected use, location, etc.).
2. Salesman gather's customer information, arranges trade-in appraisal, cost and pricing, prepares and delivers offer to customer.
[If customer does not accept offer, identify new car for potential purchase.]
3. Customer accepts offer and provides initial payment to salesman.
4. Salesman coordinates Insurance and Registration transfers from trade-in vehicle to new car.
- 5a. Insurance Rep updates customer policy to transfer coverage to new car.
- 5b. DMV transfers registration/plates to new car.
6. Salesman prepares final paperwork for signature and transfer to customer.
7. Customer finalizes paperwork, provides trade-in, receives new car.



Lower Level Process Model: Detailed tasks with inputs/outputs included for Create Offer process:



Use Cases and User Stories

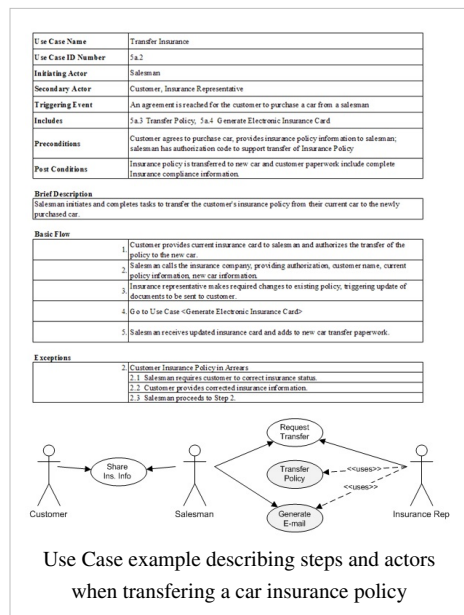
A Business Use Case or Business Scenario represents a sequence of events or circumstances that occur to accomplish a business goal. The functional requirements that are needed to accomplish the goal are embedded in the Use Case. The narrative and diagram describe how people/systems interact to achieve the 'Post Conditions'. They may incorporate only functional needs, without regard to 'how' the tasks are accomplished or system specification details may be incorporated into the Basic/Exception/Variation steps or supporting document sections.

User Stories are simple statements of atomic-level requirements that document the functions that will be developed during an Agile development methodology Sprint. They include the affected business role, business need or goal and may include the benefit to be realized with the Sprint deliverable. User Stories are expressed in the format:

"As a <role>, I want <goal/desire> [so that <benefit>]".

For the Insurance Transfer process above, a User Story that may be developed might be:

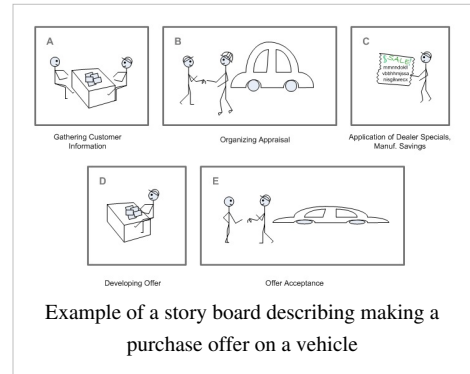
"As a <salesman>, I want <to automatically receive a new insurance card> [so that <the customer can legally drive their new car off the lot>]".



Graphical Illustration

Graphic images provide an effective tool for eliciting requirements and communicating with project stakeholders. They may closely mimic an actual or planned activity for organizing a discussion, or they may provide only theoretical User Interface rules that should be applied to the development effort.

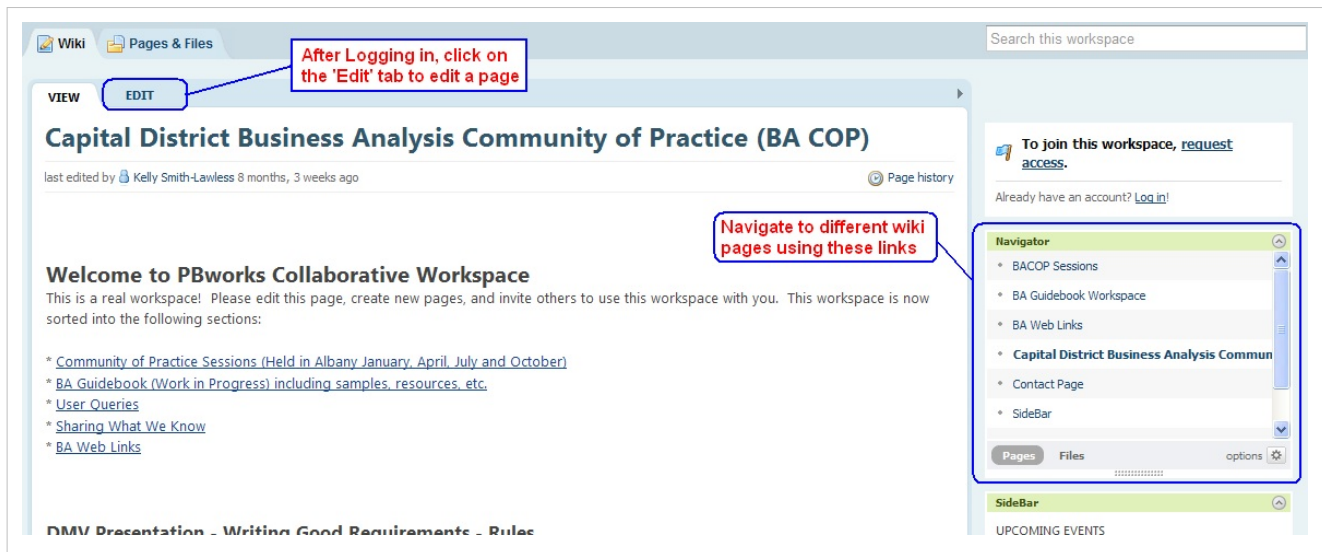
Storyboards can be used to illustrate steps in a process (see figure at right) or may be used to illustrate screen interfaces including input fields, screen states and behavior (i.e., error handling, input value validation).



Mockups and Prototypes

Mockups and Prototypes display essential features of an application before it is developed. These tools demonstrate what a system will look like, not how it will be developed. The appearance of the images or screen elements may be very close to what is the expected final version, or they may include only the barest framework showing elements (buttons, text fields, etc.) and their behavior.

Mockups can be created by editing a screen shot of an existing application. The screen print can be edited to add objects or remove objects using image-editing software such as SnagIt, MS Paint, Adobe Print Shop or another graphical editing application.



If the Mockup shown above was a prototype, someone viewing the image on a screen would be able to click on the 'Edit' tab to open the page in edit mode and click on the navigation links to open the associated pages. A prototype would demonstrate the behavior of the interface without having the full programming behind the scenes. This enhances the clarity of the business requirements, preventing defects as the planned/developing solution is reviewed by stakeholders.

Wireframes are a type of mockup or prototype that shows the framework for a web site – what will be displayed, approximate placement, field types and navigation, etc. Generic shapes are used to represent the fields and objects displayed on the proposed web page.

Data Dictionary

Data Dictionaries contain information about the data (metadata) that is stored in an application database. This tool helps to explain what the stored information represents and can be used when developing data models for an application. Generally the dictionary for a database application will include the table names, information regarding what each entity (table) represents, each field name with a definition of what is stored in the field, the formatting for the field, whether the field must be unique, whether the field is required and any default value for the field.

For an application that is used at the Department of Motor Vehicles to record information about the owners and their cars, a partial data dictionary may look like Figure n+6.

Registration - Owner Information

Owner Name:

Driver License Number:

Owner Mailing Address: ☐ Apt #

City/Town State Zip Code

Owner DOB:

Owner Phone:

Save Cancel

Example of a Wireframe screen designed to capture owner information

Data Dictionary - Owner Registration Information

Entity: Owner This table contains information about the people who own a registered vehicle

Field Name	Description	Type	Specifications	Default	Required	Unique	Key(s)
DLID	Drivers License Number	Character	9 numeric characters		Yes	Yes	PK
Last Name	Owner's Last Name	Character	25 alpha-numeric characters		Yes	No	
First Name	Owner's First Name	Character	20 alpha-numeric characters		Yes	No	
Middle Name	Owner's Middle Name/Initial	Character	25 alpha-numeric characters		No	No	
DOB	Owner's Date of Birth	Date	'MM/DD/YYYY' format		Yes	No	
DayPhone	Owner's Daytime Phone Number	Integer	10 digits; Area Code and Phone Number		Yes	No	
MailAddr1	First line of Owner's Mailing Address	Character	30 alpha-numeric characters		Yes	No	
MailAptNo	Owner's Apartment Number	Character	10 alpha-numeric characters		No	No	
MailAddr2	Second line of Owner's Mailing Address	Character	30 alpha-numeric characters		No	No	
MailCity	Mailing Address City/Town	Character	30 alpha-numeric characters		Yes	No	
MailState	Mailing Address State	Character	2 alpha characters, valid State acronym	'NY'	Yes	No	
MailZip	Mailing Address Zip Code	Character	9 numeric characters		Yes	No	
MailCounty	Mailing Address County	Integer	FIPS County Code		Yes	No	FK

Entity: Vehicle This table contains information about registered vehicles

Field Name	Description	Type	Specifications	Default	Required	Unique	Key(s)
VIN	Vehicle Identification Number	Character	14 alpha-numeric characters		Yes	Yes	PK
Year	Vehicle Year	Character	4 numeric digits, no leading zeros		Yes	No	
Make	Vehicle Make	Character	20 alpha-numeric characters		Yes	No	
BodyType	Type of Vehicle Body	Character	20 alpha-numeric characters		Yes	No	FK
OthBodyType	'Other' Type of Vehicle Body Description	Character	20 alpha-numeric characters; required if BodyType = 'Other'		Yes	No	
Color	Color of vehicle	Character	20 alpha-numeric characters		Yes	No	
Weight	Weight of vehicle (Unladen), to nearest lb.	Integer	up to 6 digits		Yes	No	
Power	Fuel source	Character	10 alpha characters		Yes	No	FK
Cylinders	Number of engine cylinders	Character	10 alpha-numeric characters		Yes	No	
MaxGrWgt	Trailer and Commercial Vehicle Max. Gross Weight	Character	30 alpha-numeric characters		Yes	No	
SeatingCap	Livery Vehicle Seating Capacity	Character	30 alpha-numeric characters		Yes	No	
Odometer	Odometer Reading at time of Acquisition	Character	2 alpha characters, valid State acronym		Yes	No	
Display	Number of Odometer Reading Digits Displayed	Integer	1 numeric character (5, 6, 7)		Yes	No	
Axles	Commercial Vehicle Number of Axles	Integer	2 numeric characters		Yes	No	
Distance	Commercial Vehicle Distance between Axles	Character	3 digit Feet, 2 digit Inches		Yes	No	

The Data Dictionary captures what data is needed, the attributes of relevant entities that participate in a process, attribute definitions and may include Entity-Relationship diagrams (see Section 3-2 Data Modeling/Data Documentation below).

Glossary

Each organization has its own acronyms, meaningful terms and specialized application of words to business processes. Creating a glossary of key business terms and definitions for a project will ensure that those involved are all communicating effectively, with a common understanding of what things mean.

A Glossary also links project work to the overall operations of the business. This tool can be useful in understanding high-level business organizational structure as well as identifying possible impacts of project work that are outside of the immediate project scope. When developing brand new solutions, a project Glossary may even be used to capture new terms that will be incorporated into the overall business jargon.

Glossaries for business projects are frequently included as tables in project documents. This allows project document reviewers to refresh their understanding of the meaning of information they may not be involved with on a daily basis.

Business Forms

Current business processes and tasks frequently rely on standard forms to collect meaningful information. Each piece of information that can be gathered using a form serves some purpose to the associated process or task. Forms provide a quick way to identify what information is important to the process or task. Exploring the 'why' of each of the fields and the relationships that may exist between different sections of a form helps to clearly identify operational requirements for a project.

Using business forms can assist in automation projects (where can this information be re-purposed from?), reengineering projects (why is this information needed and can it be removed given current constraints?) and any other type of project that is affected by where the data comes from, where it goes to and/or what can be done with its inherent information.

For purposes of registering a new vehicle, the owner information that is required by the Department of Motor Vehicles is found in section 3 of the DMV form MV-82 Vehicle Registration/Title Application

Sample box within a vehicle ownership form
when registering with a DMV

Why Use These Tools?

Use one or more of the tools listed above during the requirements gathering phase of a project to capture requirements. They help to accurately capture well-constructed requirements that reflect the actual needs and underlying concerns of the project stakeholders. Diagrams or lists that can be viewed while discussing requirements will facilitate identification of missing data, prevent the need for correction of misunderstandings, illustrate implications of proposed changes and determine specifications for the development phase.

Diagrams and other requirements gathering tools simplify communications with project stakeholders. Graphic images can clearly encapsulate many words, promoting a common understanding of what the requirements are so that they can more easily be validated. Simple illustrations and lists facilitate requirements gathering sessions by focusing the discussion and keeping everyone on the same page. Open questions can clearly be identified for follow-up and design specifications can be captured to ensure completeness.

When using diagrams and lists, requirements relating to inputs and outputs, behavior, data sources, content and other specific information can be accurately and completely documented. Documentation of the requirements will in turn facilitate the creation of design documents, test plans and scripts as well as training and user manuals.

How to Use These Tools

Focus for Requirements Facilitation/Elicitation

A review of a high-level process diagram provides a context for meeting attendees. Starting at this high level, the participants at a requirements gathering event can be acclimated to the subject of the discussion before drilling down into particulars of the process. A low-level process map can be used to identify inputs and outputs to the process, any communications that occur during the process, alternative process flows, process customers, data sources and any other of a number of elements that are needed to define requirements. Business Use Cases and User Stories provide the foundation for understanding what business task or process needs to be accomplished, allowing existing requirements to be refined and completed from the user perspective. Use Cases can encompass more than one process, capturing required interactions between actors in a process. The User Story encapsulates a single value-adding feature so that development efforts can be directly focused to the goal of the Story.

Graphical illustrations become more and more important as a way to set user expectations. These diagrams illustrate the application of requirements to a real business process and often identify missed or incomplete/inaccurate requirements. Data Dictionaries, project Glossaries and Business Forms all provide a context for discussing design specifications. Reviewing these documents with stakeholders ensures that requirements are complete, concise and accurate.

Other Uses for Tools

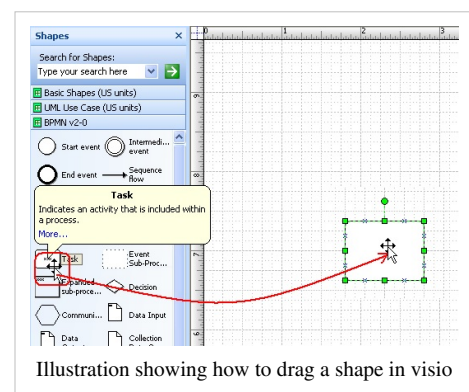
Design specifications provide direction for application developers regarding exact data, interface and process details. Mockups and Data Dictionaries are especially useful for communicating design specifications to IT developers. Data models, screen navigation and process functionalities are derived from these tools and translated into working applications.

Manuals that are created for training, administration or application users may often include mockups, diagrams and other illustrations that are developed during the requirements gathering process. Some organizations provide guidance to users at implementation that mirrors the basic flow of the Business Use Case as well as the Use Case exception or variation process flows. The requirements are incorporated into user and administrator manuals to help users in understanding why and how the established application does what it does.

How to Create Some Diagrams

Visio

One of the most popular tools available to business analysts in the public sector is the Microsoft® Office Visio® . The Visio product includes many shape stencils that can be used for many different types of diagrams. These shapes are organized for the analyst into groups reflecting the types of diagram that can be created. For example, many of the Software stencils that come with Visio support UML diagrams. There are many sources to download the OMG BPMN 2.0 Visio stencil for free on the Internet and for Visio 2010, a BPMN (v1.2) Template is included with the software.



Visio uses drag and drop to create diagrams that use shapes provided on shape 'stencils'. Connectors that are included on the stencils are used to indicate flows, relationships, messaging or other object interface connections or associations. Objects may be modified in Visio by right-mouse clicking on them. The pop-up menu displayed may include extra properties embedded in the object that can be adjusted.

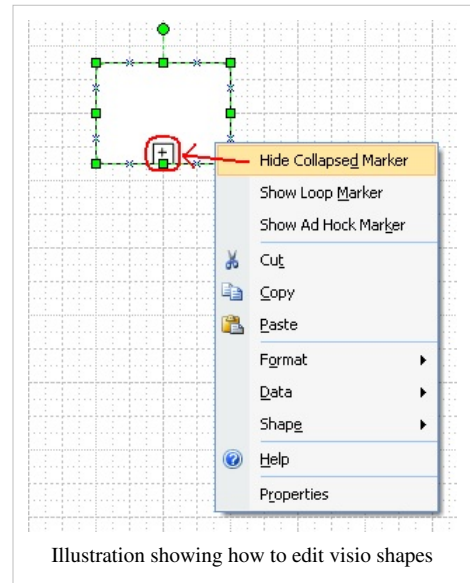


Illustration showing how to edit visio shapes

When adding connectors between objects, place the connector so that the ends of the shape are enclosed and the connected shapes are highlighted; this will ensure that when one of the shapes is moved, the connection will be preserved and the connector will automatically move to the most logical placement between the two objects.

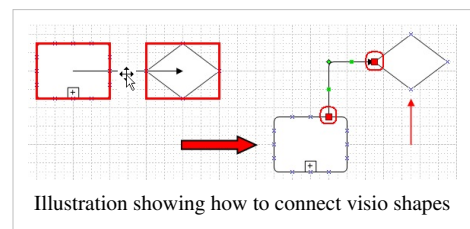


Illustration showing how to connect visio shapes

Microsoft Excel

An excellent tool to use for a variety of business analysis tasks is the Microsoft® Office Excel® application. The application supports lists, metrics formulas, charts, reporting, filtering, sorting, consolidation and many other tasks that an analyst performs. The print setup feature allows worksheets and workbooks to be pre-formatted for printing so that reviewers will be able to generate readable hard copies easily. Spreadsheet cells, data, columns and rows can be quickly formatted to accurately reflect the contents.

Microsoft Word

Document artifacts can be produced using Microsoft® Office Word®. Requirements documents are typically used to present and distribute project requirements for review and approval. Documents can be converted to .pdf documents for posting and distribution using the Adobe® Acrobat® Professional software.

General Tips

Each software application that is used to generate diagrams includes extensive documentation in the Help feature. Many specific actions are described and explained in the Help files and should be used for reference.

When copying graphic images that were creating using a tool that embeds formatting, convert the image to an unformatted image before pasting into a document or artifact file. This is done by copying and pasting the image into a format-neutral application such as MS Paint, then copying the image from the format-neutral application to the final document or artifact. This procedure ensures that all readers of the document will be able to view the image without requiring the originating software application to be installed on their computing device.

When setting up diagrams, use the source application header and footer configuration to include the file name, print date and page numbers. This will support an easy way to retrieve the originating file, identify when the diagram hard

copy was generated and help keep multi-page diagrams organized for the reader. Including the diagram's baseline date in the header/footer ties the diagram directly to the project timeline.

Business Analysis Within a Typical System Development Life Cycle

Business Analysis within typical System Development Life Cycles

Introduction

This section of the BA Handbook describes the standard phases and major processes of the System Development Lifecycle (SDLC), using a common language and in sufficient detail to provide a Business Analyst an understanding of the system development lifecycle and the expected deliverables for the various phases within a project.

All software development projects, software enhancements, or software procurements should begin with an Information Technology Investment Request (ITIR), Business Case, and/ or a Project Proposal. These requests then go through an Information Technology Governance process supported by the agency's Project Management Office (PMO). This process involves an evaluation and approval of the request at either a Division or Departmental level depending on the enterprise impact. IT Governance is the process that agencies use to ensure that IT is applying effort to the right projects and enhancements. The purpose of the process is to align IT investments with strategic goals, operational support, and required maintenance.

Once a proposal has been approved, it is added to the Project Management Portfolio and depending on resources, a Project Manager (PM) or Project Coordinator (PC), and a Business Analyst(s) (BA) are assigned to the project. This phase of the project is called the Origination Phase in the Project Management lifecycle. Software development does not begin in this phase but many of the tools and techniques used in the System Development Lifecycle (SDLC) are essential in the development of Business Cases and Project Proposals.

The role of BAs on an IT Project can be multi-fold. Project Team roles are described in detail in Figure 3-3, Roles and Responsibilities. It is possible for Project Team members to have multiple roles and responsibilities. On some projects, the BA may take on the roles of the Business Intelligence Analyst, Database Designer, Software Quality Assurance Specialist, Tester, and/or Trainer when there are limited resources available. It is also possible that the Project Coordinator, the Application Development Lead, or a Developer take on the role of the Business Analyst on some projects. There are two distinct sets of deliverables that a BA may be responsible for producing, the Project Management Methodology (PMM) deliverables, and the SDLC deliverables. The PMM deliverables may begin at project origination and end at project closeout. The PMM and the SDLC are closely integrated and both sets of deliverables are dependent upon each other. This SDLC begins during the Project Management Initiation Phase after the Business Case, the Project Proposal, the Project Charter, and the Project plan have been developed. This section of the BA Handbook will focus on the SDLC deliverables.

This overall framework for software development is based on the New York State Software Development Lifecycle from Section III of the NYS Project Management Guidebook Release 2 (CIO-OFT) 1. This SDLC is designed to be generic enough for virtually all system development efforts, and allows utilization of nearly all platforms, tools, and techniques. An overview of the SDLC phases and the different methodologies (workflow patterns) are described below.

System Development Lifecycle Methodologies (Workflow Patterns)

There are numerous System Development Lifecycle (SDLC) methodologies to choose from for system development projects. Many methodologies are driven by the application development tools, by the software architecture within which the application will operate, or by the “build versus buy” decision. The four common methodologies that are included in this section are Waterfall, Agile, Iterative, and Incremental. Project Managers select the SDLC methodology that best fits their project based on the project, project environment, project team and project manager attributes.

- **Waterfall methodology** - In the waterfall model, system design is broken down into a number of linear and sequential stages, in which system evolution is seen as flowing progressively downwards, through the phases. The waterfall model has distinct goals for each phase of development. In this development method, each phase must be completed before the succeeding phase can begin. The output from each phase forms the input to the next phase; therefore, each phase has to be accomplished in turn to maintain the linkage between the inputs and outputs. Many software development projects still use the Waterfall methodology. Attributes that would make the Waterfall method the recommended methodology are when formality is called for, when there are disparate stakeholders, or when there are changing resources.
- **Agile methodology** - Agile software development is based on iterative development that advocates a lighter and more people-centric viewpoint than traditional approaches. Requirements solutions evolve through collaboration between the customer and self-organizing project teams. Feedback, rather than planning, is the primary control mechanism. The feedback is driven by regular tests and releases of the evolving software. The frequent inspection and adaptation in the agile method encourages teamwork, self-organization, and accountability. Agile methods allow for rapid delivery of high-quality software and employ a business approach that aligns development with customer needs and agency goals. The Agile methodology matches the need for emerging technologies and development styles as well as quick delivery and decreased overhead.
- **Iterative methodology** - The iterative methodology is an enhanced version of waterfall methodology. As with the classic or linear-sequential life cycle model, iterative also maintains a series of phases that are distinct and cascading in nature. Each phase is dependent on the preceding phase before it can begin and requires a defined set of inputs from the prior phase. More so, a fair amount of time is spent in the analysis phase. After this phase of the project, the requirements are categorized based on their priorities and the target is met in finite deliverables in multiple iterations. Only a limited set of requirements is constructed through to implementation. The process then repeats itself. Each output from any given iteration could potentially serve as an input to the downstream requirements in the new and next requirements phase (for the next iteration). The Iterative methodology delivers a product faster and involves the customer more so they feel that they have contributed to its development and are satisfied more in the end. It would be beneficial particularly to larger in-house development and to some COTS projects.
- **Incremental methodology** - Iterative and Incremental methodologies are similar in that the scope of the project is defined in the charter and both are developed in phases, each phase adding additional functionality to the system. In Incremental development, the Scope, Requirements Analysis, and Design phases are performed sequentially. However, the software features/requirements are divided into multiple, independent groups. The Construction, Acceptance, and Implementation phases are then performed for each group with the resulting system being deployed to production. Each piece of functionality is incrementally delivered. Whereas, Iterative development performs only high level Requirements Analysis initially and chunks out the scope into logical iterations. Each iteration includes a more detailed Requirements Analysis, Design, Construction, Acceptance, and Implementation phase.

Though seemingly similar, the iterative and agile methodologies differ in that requirements could continuously evolve in agile where as in an iterative model, requirements are refined only during the requirement phase of each iteration. Another Iteration of the process is accomplished through implementation with the result being an

“Evolved” form of the same software product. This cycle continues with the full functionality “Evolving” overtime as multiple iterations are completed.

In reality, each phase of the SDLC can be thought of as a mini- project in itself, requiring planning, execution, and analysis. As the Project Team proceeds through the project and executes each phase, they will collect additional information that will enable the refinement of subsequent phases. Some of this information will be a natural by-product of having performed the processes associated with the current phase. As the detailed technical design evolves throughout the System Design phase, the team will have a much better understanding of the modules that will need to be built during construction, and will therefore be able to refine any prior estimates and plans for System Construction.

Additional information can be obtained through a focused analysis effort, performed at the completion of each phase. The responsibilities of the Project Team include assessing how closely the phase met Customer needs, highlighting those aspects of the phase that worked well, identifying lessons learned and best practices in an attempt to derive ways to improve upon processes executed throughout the project, and, most importantly, communicating results.

The SDLC Phases described here should be consistently performed even though the order of occurrence of when these activities take place may differ based on the methodology chosen. For a Plan driven methodology like waterfall, the SDLC phases would occur sequentially where as in Change driven methodology like Agile, these could occur simultaneously and repeatedly. HH: Many factors may influence your choice of approach to follow when developing a system. The better you know your Customers and Stakeholders, and the better you understand the factors that may influence their assessment of the project, the more likely it will be that your approach will suit their personalities, preferences, vision, and needs.

The key is to pick the approach that you believe will provide the best complete solution, balancing the preferences of your Customers, the abilities of your Project Team, and the overall business drivers (legislated timeframes, overall time to market, etc.). In any approach, the basic SDLC processes must be performed. What differs is the timing of their execution. As with the project management methodology, if processes or deliverables are skipped, the Project Team must record the reasons why, and must describe how the objectives of that process/deliverable will otherwise be met.

System Development Lifecycle

There are standard phases and processes that all system development projects should follow, regardless of environment, tools or work flow patterns. Every phase of the SDLC should include Information Security considerations. Security discussions should be performed as part of (not separately from) the development project to ensure solid understandings among the project team of business decisions and their risk implications to the overall development project. Security audits/reviews should be performed periodically throughout the systems development life cycle phases and conducted by the Information Security staff, independent of the development staff. Security audits/reviews should assess the status of information security from Origination through all the SDLC phases listed below. This section describes the six standard phases and the major processes of the New York State System Development Lifecycle (SDLC).

System Initiation Phase

This phase consists of the following processes:

- Prepare for System Initiation: The project team familiarizes themselves with the Business Case and Proposed Solution developed during the PMM Origination phase.
- Verify Proposed Solution: These documents are re-examined to ensure that they still appropriately define and address an existing organizational need.

To verify the Proposed Solution, the Project Team must:

- Understand the agency's current Strategic Plan, and how the new system fits into it i.e. any changes to the system must ultimately align to the strategic vision and goal of the organization;
- Consider how it fits into the Agency's application portfolio of existing or Proposed Systems. A System Context Diagram can be used to illustrate how the new/modified system will make use of, or add to, existing data stores. It can depict how the data/system will interface with other systems identified in the Strategic Plan (extract, update, data entry, etc.).
- Update the stakeholder map.
 - To initially assess and determine all impacted Stakeholders;
 - To identify the right Stakeholders representing each affected Program Area with their authoritative levels for requirements approval;
- Verify the proposed technology solution in view of Stakeholder needs, the Performing Organization's long term technology direction, constraints, and state-of-the-art technology;
- Define assumptions and constraints that could be technical, administrative or regulatory
- Identify different solution options;
- Confirm feasibility of the proposed system development approach.
- Determine SDLC Methodology (Work Flow Pattern) and Develop System Schedule: This verification effort provides the Project Team with the basis to determine the SDLC methodology (Work Flow Pattern) that will be used for this project. It will also provide the basis for a detailed schedule defining the steps needed to obtain a thorough understanding of the business requirements and an initial view of resource needs.
- Security planning should begin in the initiation phase by 2:
 - Identifying key security roles for the system development;
 - Identifying sources of security requirements, such as relevant laws, regulations, and standards;
 - Ensuring all key stakeholders have a common understanding, including security implications, considerations, and requirements; and
 - Outlining initial thoughts on key security milestones including time frames or development triggers that signal a security step is approaching.

This early involvement will enable the developers to plan security requirements and associated constraints into the project.

At the end of System Initiation, all system-related materials gathered and produced by the Project Team should be consolidated and prepared for use as input for the next phase.

System Requirements Analysis Phase

In the System Requirements Phase, the needs of the business are captured in as much detail as possible. The Stakeholders are defined in the Business Case and the Stakeholders Map. The requirements form the basis for all future work on the project. It is important that the Project Team create a complete and accurate representation of all requirements that the system must accommodate. Accurately identified requirements result from effective communication and collaboration among all members of the Project Team and Customers. This provides the best chance of creating a system that fully satisfies the needs of the Customers. The requirements for a system are

captured in the Functional requirements, Non-Functional requirements, and Business Rules.

During earlier phases or even during Requirements Phase when the requirements are elicited from the Customers of different Business Units, the Project Team must think about Business Process Re-Engineering. For example, what current processes must be changed or could be changed to do business more effectively. Is there a possibility to automate the process or leave it manual? Is it appropriate to create a single source of data that is being used by different Program Areas to avoid duplicity, maintain data integrity, and so forth?

By obtaining a detailed and comprehensive understanding of the business requirements, the BA can develop the Business Requirements Document (BRD). The BRD consists of the Business Rules, the Functional Requirements, the Non-functional requirements, the Process Model, and the Logical Data Model. In this phase, any applicable non-functional requirements standards that a system must adhere to are also captured and reviewed by appropriate units (such as Information Security Office, Server group, database management unit etc) for compliance. This phase consists of the following processes listed below.

Prepare for System Requirements Analysis

In this process, steps are taken to ensure that the project environment and Project Team are adequately prepared to capture and analyze the system requirements. All Project Team members must share a clear and common understanding of the scope of this phase of the project, the Project Schedule, the deliverables to be produced, and their individual responsibilities relating to the creation of these deliverables. In preparing for this phase, Skills, Technical Tools, and Team Assessments are done on the Project Team and the environment in which the team will work.

Regardless of the size of the development effort being undertaken, System Requirements Analysis may place the greatest demand upon Stakeholders in terms of resources and the extent of their required participation. Therefore, the Project Team must ensure that the Stakeholders identified initially are still correct and up to date. Stakeholders not identified lead to missing or erroneous requirements that could have a major set back to the project. As a part of preparing for this phase, the Project Team must make sure that the Stakeholder Map is current. Depending on the methodology chosen for this project (Waterfall, Agile or Iterative identified during the Initiation Phase), it is important for the Project Team to establish some expectations and agreed upon guidelines around communication and its frequency, sign off procedures, and the change control process with the Stakeholders.

Deliverables:

- Established Team and Environment for Requirements Analysis – set up the project repository, get access to database environments, get software tools loaded if not already.
- Context Diagram – a graphic design that clarifies the interfaces and boundaries of the project or process at hand. It not only shows the process or project in its context, it also shows the project's interactions with other systems and users. A Context Diagram shows the system boundaries, external and internal entities that interact with the system, and the relevant information flows between these external and internal entities.
- Stakeholder Map – a visual diagram that depicts the relationship of Stakeholders to the solution and to one another.

Determine Functional and Non-Functional Requirements

In Determine Functional and Non-Functional Requirements, information is gathered from a variety of project participants relating to the vision and scope of the system. There are a number of techniques used to capture the requirements. Depending on the SDLC Methodology used, the agency templates, and agency toolset, the Business Analyst may utilize some of the following techniques and/or tools to help document requirements and ensure that they are not missed. The Business Analyst Tools and Techniques section contains description of some of the techniques available such as: Use Case Modeling, Storyboarding, Functional Decomposition, interviews, joint application design sessions (JAD), Unified Modeling Language (UML), prototyping, data flow diagramming,

process modeling, and entity-relationship diagramming.

From this, specific detailed requirements are identified and documented so that they address the stated business need. It is very important that the dependencies and the interrelationships between the requirements be captured. The Requirements Traceability Matrix is crucial in identifying the right 'order' of requirements to be executed, and to enable an impact analysis if requirements need to be changed. It is also important that all the applicable business rules be documented separately from the process; they must be clearly written and then reviewed for any conflicts.

Functional Requirements The Functional Requirements specify the full set of functional capabilities needed by the new/modified system. Requirements that define those features of the system that will specifically satisfy a stakeholder need, or with which the Customer will directly interact. Any data or reporting requirements that a system must have will be defined here. The Functional Requirements will evolve throughout this phase of the SDLC as detailed Functional Requirements are captured, and as supporting process and data, models are created, ensuring that the eventual solution provides the Customers with the functionality they need to meet their stated business objectives.

Functional Requirements Tasks:

- Identify and describe the program areas that will be affected by the new system.
- Define in detail the procedures, high-level data requirements, existing reports and documents, roles and responsibilities, and the Stakeholders that will be impacted by the new system.
- Describe in detail the business rules and data requirements.
- Conduct a Security Impact Analysis early in the requirements phase and ensure participation with the Information Security representative. Key security activities for this phase include 2:
 - Initial delineation of business requirements in terms of confidentiality, integrity, and availability;
 - Determination of information categorization and identification of known special handling requirements to transmit, store, or create information such as personally identifiable information; and
 - Determination of any privacy requirements.
 - Describe existing computer systems, data, and existing data/system interfaces.
 - Ensure that all required data elements have been identified in the data analysis.
 - Ensure that the sources and destinations of the data are known.
 - Identify and prioritize any procedural and/or automation changes.
 - Make sure requirements that are identified align to the business goals and objectives.
- Make sure that the list of Stakeholders identified during initiation phase (who will identify/participate and/or approve requirements, along with their authority and/or influence levels) is up to date. Any new/change in Stakeholders identified during requirements elicitation phase should be updated in the stakeholder map.
- Identify requirements using MoSCoW analysis – requirements that are MUST haves, SHOULD haves (high priority and should be included in the solution, if possible) and nice to haves but not necessary. A point scale may be used to select a vendor for such requirements. COULD haves are nice to have but not necessary and WON'T represent requirements that the Stakeholders agreed not to be implemented but may be implemented in future.
- Identify requirements' attributes such as owner, status, priority, and dependencies. Establish a requirements traceability matrix (RTM).
- Make sure any ASSUMPTIONS AND/OR CONSTRAINTS regarding business processes, available technologies, solutions, timing, and budgetary restrictions are clearly documented and communicated to all Stakeholders.
- Once the requirements are captured, VERIFY the requirements for correctness as a quality control check and VALIDATE that they align to the business objectives and goals. Make sure that the requirements are consistent and that there are no conflicting requirements.
- Requirements can be further categorized into the following functions:
 - Common functions – common features and functions

- GUI functions – screen layouts and navigational requirements
- Reporting functions – report characteristics
- Business functions – impacts the business functions
- Interface functions – data exchange between the system and others
- Batch functions – Off hours processing requirements
- Security functions – authorization, roles and access privileges
- Once the requirements are validated, make sure that each requirement has a Stakeholder identified for User Acceptance Testing (UAT). This task is an important predecessor to test plans and test cases.

Functional Requirements Deliverables

- Business Requirements Document (BRD) consisting of:
 - Functional Requirements - A document containing detailed requirements for the system being developed. These requirements define the functional features and capabilities that a system must possess. Be sure that any assumptions and constraints identified during the Business Case are still accurate and up to date.
 - Business Process Model – A model of the current state of the process ("as is" model) or a concept of what the process should become ("to be" model)
 - System Context Diagram - A Context Diagram shows the system boundaries, external and internal entities that interact with the system, and the relevant data flows between these external and internal entities.
 - Flow Diagrams (as-is or to-be) – Diagrams graphically depict the sequence of operations or the movement of data for a business process. One or more of the following flow diagrams are included depending on the complexity of the model.
 - Business process flow
 - Data flow
 - Work flow
 - Business Rules and Data Requirements - Business rules define or constrain some aspect of the business and are used to define data constraints, default values, value ranges, cardinality, data types, calculations, exceptions, required elements and the relational integrity of the data.
 - Data Models: Entity Relationship Diagrams, Entity Descriptions, Class Diagrams
 - Conceptual Model - high level display of the different entities for a business function and how they relate to one another
 - Logical Model - illustrates the specific entities, attributes and relationships involved in a business function and represents all the definitions, characteristics, and relationships of data in a business, technical, or conceptual environment
 - Data Dictionary and Glossary - A collection of detailed information on the data elements, fields, tables and other entities that comprise the data model underlying a database or similar data management system.
 - Stakeholder Map: identifies all Stakeholders affected by the proposed change and their influence/authority level for requirements. This document is developed in the origination phase of the Project Management Methodology (PMM) and is owned by the Project Manager but needs to be updated by the project team as new/changed Stakeholders are identified through out the process.
 - Requirements Traceability Matrix: A table that illustrates logical links between individual functional requirements and other types of system artifacts, including other Functional Requirements, Use Cases/User Stories, Architecture and Design Elements, Code Modules, Test Cases, and Business Rules.

Non-Functional Requirements

The Non-Functional Requirements identify the technical, operational, and transitional requirements of the application as outlined below. In addition to business stakeholders' non-functional requirements, non-functional

agency IT specific standards requirements must also be captured. Agency Application Architect Units (AAU) may already have captured their recommended non-functional standards for their infrastructure. Agency Data Management Units (DMU) may also have recommended standards for implementations of software in their database environment. Both the AAU and DMU standards should be verified that they are still current. Non-functional Stakeholder requirements must be clear and testable.

For COTS Acquisition Requirements Analysis, the Customers and Consumers' needs should be captured in the Non-Functional Requirements document as well as those to satisfy the NYS Policies, Standards, and Guidelines listed below. The specific agency non-functional standards are included in the RFP and are used to define the current agency environment. Both the AAU and DMU standards should be verified that they are still current prior to release of a Request for Proposal (RFP). It is not the intention of the RFP to prescribe the COTS technical solution but only describe the existing infrastructure. It is up to the vendor to describe how their solution will fit into the agency environment and the necessary costs associated with their product to make it work. It is recommended that pre-established structural criteria are not appropriate to handle the highly volatile COTS marketplace. Requirements that may eliminate viable COTS solutions should be avoided.

Technical Requirements

Technical Requirements identify the technical aspects and constraints that must be considered when defining the new system. Considerations may include accessibility needs of Customers and Consumers, whether or not the storage and handling of data must follow specific encryption regulations, or whether the system will operate on internal agency hardware or will be hosted at either an internal or external data center. Areas to be addressed include:

NYS Policies, Standards, and Guidelines - This requirements section defines the NYS policies, standards, and guidelines that apply to the system. The State of New York Enterprise IT Policies may be found at the following website: <<http://www.its.ny.gov/tables/technologypolicyindex.htm>>

- Web site adherence to the NYS Information Technology Policy S05-002 (State Common Web Contact Page). S05-002 can be viewed at: <<http://www.its.ny.gov/policy/s05-002/NYS-S05-001.pdf>>.
- User interface adherence to the NYS Information Technology Policy NYS-P08-005 (Policy on Accessibility of Web-based Information and Applications). NYS-P08-005 can be viewed at: <<http://www.its.ny.gov/policy/NYS-P08-005.pdf>>.
- Web user interface guidelines from the NYS Information Technology Policy G02-001 (Guidelines for Internet Privacy Policies). G02-001 can be viewed at: <<http://www.its.ny.gov/policy/NYS-G02-001InternetPrivacyGuideline.pdf>>.
- Web user interface guidelines for the NYS Information Technology Policy P03-001 (NYS Directory Services - Directory Account Management). P03-001 can be viewed at: <<http://www.its.ny.gov/policy/NYSTechPolicyP03-001.pdf>>.
- Conformance to NYS Information Technology "NYeNET" secure information network Terms of Service. The URL for information on NYeNET is <<http://www.its.ny.gov/nyenet>>.
- Conformance to the NYS Department of Homeland Security and Emergency Services (DHSES), NYS ITS Enterprise Information Security Office (EISO). policies applicable to computer systems as prescribed in the New York State Information Security Policy - Cyber Security Policy P03-002 <<http://www.dhSES.ny.gov/ocs/resources/documents/Cyber-Security-Policy-P03-002-V3.4.pdf>>.
- Accessibility - the degree to which a system is available to as many people as possible. Accessibility can be viewed as the "ability to access" and benefit from some system or entity. Accessibility is often used to focus on people with disabilities or special needs and their right of access.
- Encryption - whether or not the storage and handling of data must follow specific encryption regulations.
- Hosting - whether it is required that the system will operate on internal agency hardware or will be hosted at either an internal or external data center.

- Environment for System Operation and Maintenance – physical environment in which the system must function. Location (indoors, outdoors, residency, main office, etc.), number of locations, temperature and climate constraints, dimension constraints, stability, mobility, safety and durability are some of the factors to consider.
- Disaster Recovery/ Recoverability/ Business Continuity – addresses the ability to recover from power failures, lost data, system failure, acts of nature or sabotage. Discussion points include criticality of the system, acceptable response time to recover, point of restoration, redundancy and failover plans. Disaster recovery is a subset of business continuity.
- Archiving/Retention Procedures – defines the process to retain data after it has served its usefulness. Should data be purged from the system? How long should data be saved? Will there be a need to access historical data? How fast do you need to get it?
- Security Procedures - Security Analysis begins with an identification of the security decision makers, the systems administrator, the “delegated administrators” and the general system users. These authorization levels are defined in detail in the NYS ITS Enterprise Information Security Office (EISO) Security Activities within SDLC Phases (ITS-S13-XXX). A major consideration during Security Analysis is identification of data restrictions and requirements based on ownership, intellectual property, privacy, confidentiality, and accuracy.
- Legal, Regulatory, Protection, and Functional Security Requirements are captured and analyzed.
- Data Integrity – addresses currency of the data, accuracy, referential integrity, calculations, conversions, dependencies, etc.
- Technology Guidelines, Regulations, and Constraints – procedures and processes that need to be addressed. Industry standards, certifications, coding standards, testing standards and documentation standards, etc.

Operational Requirements

Operational Requirements specify any administrative constraints or expectations that must be supported by the system. These requirements may include system performance expectations, technical infrastructure constraints, security mechanisms that must be followed, the need to regularly archive data, and any mandated audit and control processes. Areas to be addressed include:

- System Performance and Responsiveness Expectations - Make sure any Stakeholders’ requirements about the expected response times for refresh of screens, data saving, reloading, etc. are clearly captured. Any performance requirements or system availability requirements must be documented along with any Stakeholders’ expectations.
- System Availability Requirements- when does the system need to be available, daily 7am – 5pm, 24x7, etc., maintenance window, dependencies on other interfaces, restore and reactivate procedures (Recovery Point Objective (RPO), Recovery Time Objective (RTO), etc.
- Business Continuity - requirements to ensure that critical business functions will be available to customers, suppliers, regulators, and other entities that must have access to those functions. These activities include many daily chores such as project management, system backups, change control, and help desk. Business continuity is not something implemented at the time of a disaster; Business Continuity refers to those activities performed daily to maintain service, consistency, and recoverability.
- Customers/ Consumers for the System - It is important to know the maximum number of users for the system and the number of concurrent users. Identify if the application will be used internally or externally by field users (i.e. bridge inspectors) or by different external departments (i.e. Thruway, DEC, DMV).
- Volume of Data - It is important to know the volume of data to be stored, updated, and/or reported on, the frequency of updates, what archiving of data is necessary and anticipated growth per year in data.
- Fault Tolerance/ Robustness – addresses how the system will respond to data exceptions, system failures and hardware failures.
- Data Archival/ Retrieval - Retention Period and Archiving Requirements, requirements to retain data after it have served its usefulness. Should data be purged from the system? How long should data be saved? Will there be a need to access historical data? How fast do you need to get it?

- Audit and Control/ Audit Procedures – addresses the need to trace and log use of the system as to updates in database, operations performed, web page visitors, etc.
- Software Quality Assurance (SQA) – assurance that the system meets specified requirements and Customer needs and expectations
- Activities Related to Administration and Maintenance of System/Data – addresses how authorization is assigned and process by which problems are reported and resolved.
- Compatibility with Other Applications/ Interoperability – need for system to interface with other systems without interfering with the operation of those systems. Web services, transmission protocols, messaging, data exports, data imports, and compatibility are some of the requirements addressed.
- Configuration Requirements – addresses ease of system to adapt to new functionality without the need to make coding changes.
- Manageability/ Maintainability Requirements – addresses support logistics, process for reporting incidents, change request process, routine diagnostics, defect tracking, upgrade process and schedule, etc.
- Scalability - (horizontal, vertical): operational scalability including support for additional users or sites, or higher transaction volumes.
- Reliability – What is the defect rate or failure rate of the system? What is an acceptable recovery time?
- Portability – How easy is it to migrate the system to other platforms or operating systems.

Transitional Requirements

Transitional Requirements define the realm of conditions that must be satisfied prior to physically implementing the system in a production environment, or to relegating support responsibilities to the Performing Organization. Data conversion requirements, development, delivery of Consumer training programs and materials fall into this category. Areas to be addressed include:

- Historical Data Cleansing, Conversion, and Import into the New System – addresses need to convert legacy data into new system.
- Requirements Associated with Validation of the System Prior to Release – addresses availability of customers for testing, what type of deployment is acceptable (all or piecemeal), customer expectations of system, etc.
- Roles of Customers/ Consumers of the System– needed to identify training needs, documentation needs, system access, hardware needs, deployment logistics, etc.
- Expectations for User/Technical Documentation – target audience, format and content, media, etc.
- Expectations for User/Technical Training and Training Materials - target audience, format and content, media, etc.
- Mechanics of Physically Deploying/Transitioning the Application – business process changes, big bang or soft rollout, pilot, etc.
- Required Support Hours and Acceptable Maintenance Windows - when does the system need to be available, daily 7am – 5pm, 24x7, etc., maintenance window, dependencies on other interfaces, etc.
- Monitoring: Application Level Monitoring must be explicitly requested; otherwise, you just get system and database monitoring.
- Future Development Considerations –
 - Localizability - ability to make adaptations due to regional differences
 - Modifiability or extensibility - ability to add (unspecified) future functionality
 - Evolvability - support for new capabilities or ability to exploit new technologies
 - Composability - ability to compose systems from plug-and-play components
 - Reusability—ability to (re)use in future systems

Non-Functional Requirements Deliverables:

- Business Requirements Document (BRD) consisting of:
-

- **Non-Functional Technical Requirements** – A section of the document that captures the technical requirements the system must possess. It identifies technical aspects and constraints that must be considered when defining the new system. Considerations may include accessibility needs of Consumers, whether or not the storage and handling of data must follow specified encryption regulations, whether the system will operate on internal agency hardware, or will be hosted at an internal or external data center.
- **Non-Functional Operational Requirements** - A section of the document that captures the operational requirements the system must possess. It specifies any administrative constraints or expectations that must be supported by the system. These requirements may include system performance expectations, technical infrastructure constraints, security mechanisms that must be followed, the need to regularly archive data, and any mandated audit and control processes.
- **Non-Functional Transitional Requirements** - A section of the document that captures the transitional requirements the system must possess. It defines the realm of conditions that must be satisfied prior to physically implementing the system in a production environment. This includes the transference of support responsibilities to the Performing Organization. Data conversion requirements, and development and delivery of Consumer training programs and materials fall into this category.
- **Requirements Traceability Matrix:** - A table that illustrates logical links between individual non-functional requirements and other types of system artifacts, including other Non-Functional Requirements, Use Cases/User Stories, Architecture and Design Elements, Code Modules, Test Cases, and Business Rules.

Requirements and SDLC Methodologies

There are numerous SDLC Methodologies (Work Flow Patterns). Regardless of the methodology chosen, the goal is to capture requirements as accurately as possible, the level of requirement details differ greatly depending on the stage you are in within a particular methodology.

If you are using Waterfall methodology or acquiring a COTS application, the primary goal of this phase, is to create detailed Functional and Non-Functional Requirements upfront without a need for redefining requirements at a later stage or phase.

Iterative and Incremental methodologies are enhanced versions of Waterfall methodology. As with the linear-sequential life cycle model, Iterative maintains a series of phases that are distinct and cascading in nature. Each phase is dependent on the proceeding phase before it can begin and requires a defined set of inputs from the prior phase. A fair amount of time is spent in the System Requirements Analysis Phase. After this phase of the project, the requirements are categorized based on priorities and a finite set of deliverables in multiple iterations are established. A limited set of requirements are constructed and implemented, and then the process repeats again. Each output from any given iteration potentially serves as the input to the downstream requirements in the next iteration. As in the Waterfall methodology, the requirements are locked down once this phase completes for a given set of requirements. Typically, there is an established and agreed upon change control process, which allows for changes in requirements.

Although Iterative and Agile are similar, they differ in that the requirements in Agile methodology continuously evolve. The left over requirements are given consideration along with the new functionality for the next iteration, which form the product backlog. The requirements in Agile are prioritized and re-prioritized by the product owner(s) before each iteration or sprint. The multiple iterations continue until all functionality has evolved into a complete system. Usually, there is no separate change control process as the high-level requirements captured initially are continuously elaborated.

For a COTS Acquisition, all the Functional and Non-Functional Requirements are captured before a Request for Proposal (RFP) is sent out. The System Requirements Analysis Phase for COTS is similar to the above-mentioned methodologies, such as defining the business need; identifying Stakeholders captured in the Business Case and Stakeholders Map, defining data requirements, and system functionality all need to be documented in the Functional and Non-Functional Requirements.

The following diagram illustrates some representative categories of requirements that the team consider when defining tasks and activities throughout the system development project.

EDIT NOTE: Insert Diagram Figure 3-1 System Requirements Analysis Considerations

Define Process Model

Define the Process Model may begin at any time after the Project Team has started collecting specific Functional and Non-Functional Requirements. The resulting Process Model of the system, also often referred to as the “To Be” model, illustrates the system processes as they are envisioned for the new/modified system. Over time, this pictorial top-down representation of the major business processes will be decomposed into manageable functions and sub-functions until no further breakdown is possible. When combined with the detailed set of Functional and Non-Functional Requirements and the supporting Logical Data Model, this Process Model should completely address not only the full list of business needs to be satisfied by the new/modified system, but also the vision for how the new/modified system will provide and support this functionality.

Remembering that much of System Requirements Analysis is iterative, the Project Team must ensure that as requirements are updated as a result of continued efforts to determine Functional and Non-Functional Requirements, the Project Team must also refine the Process Model to accommodate those changes.

The deliverable is a Process Model: A graphical representation of the decomposition of all business processes that interact with the system.

Define Logical Data Model

A logical data model captures the data that supports the processes and business rules - it identifies entities and their relationships to other entities, and defines attributes with their business definitions. A Database Designer with the help of a Business Analyst is responsible for designing the logical representation of the data to support the business need. A Data Dictionary and Glossary must be created during development of the Logical Data Model. This will also serve as a reference for a common understanding of business words and meanings for all the Stakeholders. Typically, this model will evolve throughout the iterations of capturing and documenting the Business Requirements. This becomes the foundation of the data repository (or Physical Data Model) and is the basis for the DBA to create the physical database, it is important that the Data Dictionary is clear in its definitions, and that all the data has been modeled appropriately.

Deliverables:

- Logical Data Model – Diagrams and Data Dictionary that define data elements, their attributes, and logical relationships as they align within a Business Area, with no consideration yet given to how data will be physically stored in actual databases.
- Data Dictionary - A collection of detailed information on the data elements, fields, tables and other entities that comprise the data model underlying a database or similar data management system
- Existing File Descriptions - Existing file layouts and existing database descriptions are accumulated and reviewed. Identification of data used to initialize the new application as well as data sources that are inputs to the normal processing cycle of the new application are compiled. If incomplete documentation is available for the existing data, analysis is done to determine the business need of the available data.
- Data Conversion Requirements – Requirements that are applicable for the data transfer between the existing system and the new system. These requirements are temporary in nature in that once the transition from existing to the new system is complete, they are no longer needed.
- Archiving and Retention Requirements – Businesses retain documents not only for Business Intelligence purposes but also for compliance requirements. Retention of documents involves systematic archiving, keeping all the different versions, with each one an exact copy of the original document. This is different than “backing up” which may mean only keeping the most recent copy. The intent of back up procedures is for more a

disaster-recovery precaution than a document-retention practice.

Reconcile Functional and Non-Functional Requirements with Models

At this point in the System Requirements phase, the Project Team ensures that the Process and Logical Data Models accommodate all business rules. An analysis is done to validate and cross-reference all requirements to the Process and Data Models. A walkthrough and review of the Requirements, the Process Model, and/or the Logical Data Model is held with the Stakeholders.

Deliverables:

Validated Functional and Non-Functional Requirements and Models – An updated set of Functional and Non-Functional Requirements, Process and Logical Data Models that have been modified to address any gaps or weaknesses identified during the review of these documents.

System Design

In the System Design phase, the Project Team builds upon the work performed during System Requirements Analysis, and results in a translation of the functional and non-functional requirements into a complete technical solution. This solution dictates the technical architecture, standards, specifications, and strategies to be followed throughout the building, testing, and implementation of the system.

This phase consists of the following processes and deliverables:

- Prepare for System Design, where the existing project repositories are expanded to accommodate the design work products, the technical environment and tools needed to support System Design are established, and training needs of the team members involved in System Design are addressed.
- Define Technical Architecture, where the foundation and structure of the system are identified in terms of system hardware, system software, and supporting tools, and the strategy is developed for distribution of the various system components across the architecture. This Technical Architecture document is created by the Application Architect with the assistance of the Business Analyst and the Application Development Lead.
- Define System Standards, where common processes, techniques, tools, and conventions that will be used throughout the project are identified in an attempt to maximize efficiencies and introduce uniformity throughout the system. These standards can be broken down into three categories:
 - Technical Development standards - describe naming conventions, programming techniques, screen formatting conventions, documentation formats, and reusable components. These may be established for all projects in a large data processing/IT shop, or may be developed uniquely for a particular project. In addition, they may be unique to a development team or industry- standard and universally accepted.
 - Configuration Management standards - provide the basis for management of the development of individual software components of the system. These standards ensure that functions such as controlling and tracking changes to the software being developed, along with backup and recovery strategies, are inherent in the development process.
 - Release Management standards - establishing at this point in the lifecycle ensures that the right level of planning occurs surrounding both the initial and subsequent releases of the system to the Customers and Stakeholders. These standards are also necessary for successfully managing migrations of the application to the various testing environments.
- Create Physical Database, where the actual database to be used by the system is defined, validated, and optimized to ensure the completeness, accuracy, and reliability of the data. The database architect creates the physical database using the physical database model based on the logical model created in the system requirements phase. The business analyst may assist in the design. When designing the database, it is important to accurately estimate anticipated data usage and volumes. Answers to basic questions will help determine the most efficient database schemas, and will enable the team to optimize the database to achieve desired

performance.

Sample considerations include:

- Expectations surrounding the use of the data.
 - The number of users expected to access the data simultaneously during normal usage.
 - Anticipated peak user loads on the system.
 - The need for audit trails.
 - Data retention needs (e.g., is it necessary to save specific details of each record, or is it sufficient for historical purposes to simply maintain summarized data?).
 - Multiple environments may be needed for development, testing, quality assurance, and production of the system
- Prototype System Components, where various components of the solution may be developed or demonstrated in an attempt to validate preliminary functionality, to better illustrate and confirm the proposed solution, or to demonstrate “proof-of-concept.”
 - Produce Technical Specifications, where the requirements of the system are translated into a series of technical specifications for all components of the system, setting the stage for System Construction. The Business Analyst plays a crucial role in the development of the Technical Specifications document. Designing a complete solution means considering each aspect of the requirements and designing a set of Technical Specifications that supports all dimensions of the project. The Technical Specifications should be detailed enough to provide all of the necessary information to the developer that they should be able to start – and complete – the assignment without any further questions. Some of the numerous aspects included in this document are:
 - Detailed module specs for all system components, whether they are common routines, GUI elements, report and batch functions, or interfaces;
 - The approach for implementing the security strategy (defined in the Technical Architecture) throughout each module of the system;
 - System performance expectations and a strategy for meeting them given anticipated system usage and peak processing loads;
 - Test Plans and Cumulative testing strategies enabling validation of all levels of the application from individual modules through a completely integrated system;
 - A complete data conversion approach addressing the cleansing and loading of historical data as well as population of new data not currently available in any existing system;
 - Documentation and training strategies aligned with the overall application, enabling development of comprehensive training curricula, and supporting materials; and
 - Deployment plans addressing the distribution and transition of the system that can be reviewed with and validated by the Consumers.

Key security activities for this phase include 2:

- Conduct the risk assessment and use the results to supplement the baseline security controls;
 - Analyze security requirements;
 - Perform functional and security testing;
 - Prepare initial documents for system certification and accreditation; and
 - Design security architecture.
-

System Construction

In the System Construction Phase, the Project Team builds and tests the various modules of the application, including any utilities that will be needed during System Acceptance and System Implementation. As system components are built, they will be tested both individually and in logically related and integrated groupings until a full system test has been performed to validate functionality. The documentation and training materials are also developed by the Business Analyst during this phase.

This phase consists of the following processes:

- Prepare for System Construction, where the system development and testing environments are established, and where the Project Team is instructed in the processes and tools that will be used throughout this phase.
- Refine System Standards, where standards established in System Design are enhanced and adjusted as the Project Team becomes more familiar with the project environment, or in response to changes in the strategic or technical direction of the project.
- Build, Test, and Validate (BTV), where individual system components and utilities are constructed and tested to ensure that they perform to the Functional and Non-Functional Requirements. Data conversion utilities are also written during this phase.
- Conduct Integration and System Testing, where logically related components of the system are assembled and tested as single units, and a complete end-to-end system test is performed.
- Produce User and Training Materials, where all User-related documentation and training materials are developed.
- Produce Technical Documentation, where all materials intended for the Project Team of individuals ultimately responsible for the on-going maintenance and support of the system are created. Some the required documentation identified in the Requirements phase may include the Updated Technical specifications, Integration Plan, Maintenance Manual, Operations Manual, System Administrator Manual, Help Desk Scripts, etc.

System Acceptance

In the System Acceptance Phase, the focus of system validation efforts shifts from those team members responsible for developing the application to those who will ultimately use the system in the execution of their daily responsibilities. Typically, these would be the business area stakeholders also identified initially during Project Initiation Phase. This phase of the SDLC is important because it is the last time that rigorous testing will be performed on the system before it goes into production. It is also very likely the first time that Customer will be able to exercise the application in a near-production environment, which adds a unique perspective to the testing efforts. In addition to confirming that the system meets functional and non functional user expectations, activities are aimed at validating all aspects of data conversion and system deployment.

This phase consists of the following processes:

- Prepare for System Acceptance, where the system acceptance environment is established, and where the testing team is instructed in the use of processes and tools necessary throughout this phase. Preparation of both the testers and the environment in which they will operate is crucial to the success of this phase. User and training materials must be distributed in advance of this effort, and any training sessions needed to familiarize the testers with the application must be conducted. The User Acceptance Test Plan must be reviewed with the acceptance testing team to ensure a clear understanding of expectations and outcomes during this phase.
- Validate Data Initialization and Conversion, where the processes and utilities used to populate the system data-base are tested to confirm that they provide a starting point from which the new system can begin processing. Confirmation before the system begins production that all utilities and processes needed to load data into the system work correctly, and that any data carried forward from another system is migrated completely and accurately is crucial to project success. A comprehensive set of completed test plans identifying all data initialization and conversion tests that were performed, along with the detailed outcomes of these tests, the list of defects identified as a result of these tests, and the results of any subsequent retests. These test results are

contained in the Acceptance Test Plan section of the Technical Specifications.

- Test, Identify, Evaluate, React (TIER), where the system functions and processes undergo a series of exhaustive acceptance tests to validate their performance to specifications, and where examination of test results determines whether the system is ready to begin production.
- Refine Supporting Materials, where the various materials that support the use, operation, and maintenance of the system are updated to reflect any necessary adjustments resulting from acceptance test results.

Key security activities for this phase include:

- Integrate the information system into its environment;
- Plan and conduct system certification activities in synchronization with testing of security controls; and
- Complete system accreditation activities.

System Implementation

The System Acceptance Phase, the final phase of the lifecycle, comprises all activities associated with the deployment of the application. These efforts include training, installation of the system in a production setting, and operational use by the end users. Transition of responsibility of maintenance for the application from the Project Team to the application support team occurs.

This phase consists of the following processes:

- Prepare for System Implementation, where all steps needed in advance of actually deploying the application are performed, including preparation of both the production environment and the Customer communities.
- Deploy System, where the full deployment plan, initially developed during System Design and evolved throughout subsequent System Development Lifecycle Phases, is executed and validated.
- Transition to the Support Team, where responsibility for and ownership of the application are transitioned from the Project Team to the unit in the Support Team that will provide system support and maintenance.

The following diagram illustrates every phase, process, and deliverable in the system development life cycle.

EDIT NOTE: Insert SDLC Chart across different project phases

Software Quality Assurance (SQA)

Software quality assurance provides the foundation on which all system development activities should occur so that the highest quality system possible will be delivered. According to the IEEE Standard Glossary of Software Engineering Terminology, quality is defined as the degree to which a system, component, or process meets specified requirements and Customer needs and expectations.

Software Quality Assurance programs should be comprised of three components – quality standards, quality assurance processes, and quality controls.

- Software Quality Standards define the programming standards, and development/testing standards to be followed throughout the project.
- Software Quality Assurance Processes define practices and procedures to be used by the Project Team to meet the quality standards, and to provide management with evidence that these procedures are being followed.
- Software Quality Controls comprise a series of reviews and audits that evaluate deliverables with respect to defined standards and acceptance criteria. These controls include software testing techniques and peer reviews.

SQA efforts must be performed throughout all phases of the project. Ideally, there should be a separation of duty from the team members responsible for delivering the system and those responsible for quality assurance; unfortunately, many agencies do not have the resources to provide an independent unit to perform these tasks. In many instances, these tasks will be performed by the Project Team.

Project Roles and Responsibilities

When staffing system development projects, there are a number of roles that should be considered. Team members may be tasked with several roles. The roles identified within the SDLC are representative of those that are typically required in a system development effort and are described in Figure 3-3.

EDIT NOTE: Insert Roles & responsibilities chart here

SDLC at a Glance

The following table lists all SDLC phases' processes, some techniques available for use in executing these processes and process deliverables (outcomes).

EDIT NOTE: Insert SDLC at a glance chart here - incl tools & deliverables

References

^ New York State Office for Technology, 2003, The New York State Project Management Guidebook, Release 2, Section III: System Development Life Cycle, <http://www.cio.ny.gov/pmmp/guidebook2/index.htm>

^ NIST Special Publication 800-64 Revision 2, Security Considerations in the System Development Life Cycle, October 2008

Data Modeling and Data Documentation

Business data is a critical asset of any organization. The information contained within data plays a large role in the operations, reporting, and measurement of how the organization is performing. An information system provides the means to capture, share, and monitor business data. A sound data structure is universally required to support business activities and to reduce future costs. The need to construct the 'right' data store for an information application involves data governance, understanding data flows, and developing a reliable foundation to ensure reliability, stability, and maximal effectiveness for the business investment.

Business Data

Data Governance and Security

Investments in information systems that are necessary to build new applications, modify existing applications, or migrate to COTS (Commercial Off-the-Shelf) solutions can be very high for any organization. In order to ensure that those investments result in maximum returns, controls may be implemented in the form of data governance and security. Controls may take many forms and may be established at the organizational level or below.

Data governance is concerned with who is responsible for the organizational data and how governance decisions and processes are executed. One or more different approaches may be used in the governance of business data. These include:

- Top-down – decisions affect the entire organization, authoritative
- Bottom-up – decisions relating to data established at low level (e.g., naming conventions)
- Center-out – typically consultant hired to consider governance needs from all stakeholders and make organization-wide recommendation (results in Top-down for approved decisions)
- Silo-in - multiple groups set governance standards based on collective agreement [1]

Goals of data governance include ensuring the quality and security of business data, conformance of the data to organizational standards, overarching data rules and definitions, and the assurance that the data will be trustable.

Business Ownership

When creating a new information system, the application is generally sponsored by one or more business program areas or departments. Organizational data governance requires the identification of those who will 'own' the data. These owners will be responsible for oversight, coordination with the organization's Governance Board, and securing the data access rights. An 'Owner' may be responsible for data within the domain of the application, or the responsibility may be spread out between multiple owners across all applications that the data will interact with[2]. Regardless of how the owner (or owners) of the data is determined within an organization, the approvals for project data requirements will come from within this ownership role.

Data Documentation

Documentation of application data is critical to ensure a common understanding of the information contained within the data and to assure the reliability of that information. Business Analysts are often responsible for creating data documentation for information system projects and the alignment of this documentation with governance processes and rules. Documentation artifacts will span the entire SDLC from Initiation through Implementation.

Modeling Data Requirements

Logical Entity Relationship Model

^ 1. Data Governance Institute, **Choosing Governance Models**, retrieved 02/05/2014 from http://www.datagovernance.com/gbg_governance_models.html.

^ 2. Thomas, Gwen, **Assigning Data Ownership**, retrieved 02/05/2014 from http://www.datagovernance.com/gbg_assigning_data_ownership.html

References

- [1] http://en.wikibooks.org/wiki/Business_Analysis_Guidebook/Data_Modeling_and_Data_Documentation#endnote_Gov._Models
- [2] http://en.wikibooks.org/wiki/Business_Analysis_Guidebook/Data_Modeling_and_Data_Documentation#endnote_Data_Ownership

System Acceptance, Test Planning and Strategy

System Acceptance, Test Planning and Strategy

The Testing Role and Responsibilities

The primary role of the tester is to ensure that development work is implemented in production with as few defects as possible- hopefully defect free! The leading contributor of ensuring a successful implementation is communication. The tester uses tools to facilitate planning and performing the required tasks. The Test Strategy, Test Plan, UAT Checklist and Defect Tracking documents are the tools used to assist in a successful implementation. Each document serves a specific purpose in communicating the testing needs to the project team.

Defect Tracking is primary task used for communicating testing results. Most IT shops have an OTS (off the shelf) product that serves as a repository that records, reports and provides metrics on defects. In the absence of a defect management product, and if a project is small enough, a spreadsheet can be used to capture defects and their resolution. Each of the tester's tools will be discussed below.

EDIT NOTE: Insert into margin....Although testing, test planning and the development of test cases is usually performed by the Business Analyst, these functions will not earn experiential credit towards the IIBA CBAP or CCBA certifications. Review the certification requirements in the IIBA Handbooks at www.iiba.org.

Test Planning - The Test Strategy Document

Test planning involves determination of the what, when and how- testing is performed. The Test Strategy captures and documents everything that should be tested. The Test Strategy is a high-level plan which identifies all system and processes that should be considered. It is the tester's responsibility to write up the Test Strategy, but it should be presented to the team for feedback, and to the team leads for final approval.

The Test Strategy identifies who is responsible for testing the different system components within the stages of development. It includes the date the product is expected to be received from the development team and ready to begin testing. The Test Strategy captures the length of time it will be tested and also identifies what tools and methods will be used to confirm the expected results.

The document is also a communication plan as related to testing. The Test Strategy identifies the responsible parties for testing, approval, and user acceptance testing (UAT). Team members need to track and have confidence that defects are fixed and retested before release to production. Testers and other team members need to have assurance that the Test Plan is sufficient, correct, is verifiable and that other systems are not negatively affected.

Developing the Test Plan

The Test Plan and Test Strategy are separate and different documents. The Test Plan is developed once the business rules and functional specifications are complete. It may be created by the tester, programmer or collectively by both. The Test Plan identifies the storage location of the Business Rules and Functional Specifications and any additional reference documents. The Test Plan will identify the change being tested. In addition it contains:

- The Business Rule to be tested
 - The Condition to be tested
 - The Activity performed to run the test
 - Test Record data needed to perform the test
 - The Expected Results of the test
 - The Actual Results of the tests
-

The tester has a unique role, in that they have technical knowledge of the system, but also have the perspective and insights of a non-technical user. A good Test Plan will test the proper technical function of the business rule, but also perform functions that are not in the normal path. Non-technical users use systems in ways that are unexpected and unplanned.

Priority Testing

All business rules should be tested, but sometimes there isn't enough time to test each and every possible scenario on every business rule. Risk assessment and prioritization is part of the considerations for developing the test plan. Testing the cases and systems that have the risk with the highest cost of failure should be given priority. In prioritizing the test cases, professional judgment should be used, and feedback from the developers and approval from business unit decision-makers should be obtained.

Regression Testing

The tester also needs to ensure that the system has not been changed or affected in unexpected ways. Regression testing is performed for this function. For example, when a single module is updated and should affect only a certain transaction or flow, the tester should test that other unchanged transactions that share that module still function as they did before the change was implemented.

Regression testing is a form of testing that ensures that the processes that should not have been changed are not. It is a critical part of system testing and should be part of the Test Strategy planning. Automated testing tools are used mainly for this purpose, such as HP Quick Test Professional (QTP). The automated tests are called scripts and are developed before any changes have been migrated to the test region. The scripts should be developed to run through a large variety of test cases. When they are developed, they record the expected output as a baseline of the system. When running regression scripts for integration testing, the tester can easily identify unexpected results in the regression test analysis and then report on the found bugs accordingly. Scripts will need to be re-recorded and / or re-baselined as a result of program changes.

Positive / Negative Testing

This type of testing perspective looks at a scenario that ensures a certain test does (positive) or does not do what is expected (negative). For example, if we want to test a scenario where entering a zip code in an online form will populate the city and state fields automatically after hitting enter, our test case may say "Enter the zip code '12208', with an expected result of Albany, NY." A negative test case would say "If the user enters a valid zip code, the Error Message 'Invalid Zip Code Entered' will not display."

Positive and Negative test cases are important because they ensure that the testing performed on a single business rule is ensuring the process is working as expected when it is performed correctly and also that it is working as expected when the transaction/process is not working within a valid test path. The error paths need to perform the correct functions when the correct path is not followed.

Volume / Load Testing

Other forms of testing include Volume and Load testing. Volume testing involves a check of the system's response to requests when the system has large amounts of data. For example, if a user wanted to perform a complex query against a database that has a significant quantity of records, the query may bog down the system and cause it to take a longer time returning results than if the query were simple. Load testing involves a test of the system's capability of handling high demand situations. If the traffic is at its peak, the load test will measure how the system handles the condition. For example, if there were a great bargain online the large number of concurrent users trying to access the shopping cart at once will put demand on the system. The demand from many users is a Load test. Load testing is also considered 'Stress' testing.

Environments

The tester is usually the gatekeeper for code migration coordination. Programmers develop and test in their own region, usually called the development region. The developer is responsible for unit testing the code in the development environment to ensure it works properly. As code is tested and approved, it moves through the test region and eventually to the production region.

System/integration testing is usually performed in a separate test environment staged between development and production. The test region should be a copy of production along with any programming changes that have not yet been migrated to production. Some IT shops may have additional regions such as Training and Volume and even a Pre-Production region. Each region is for specific testing needs of the IT shop. For example, an entire duplicate region of production may be kept for the sole use of running ad-hoc reports. The purpose of keeping a separate region for this purpose is to provide the needed information for users and also to keep the load off of the production region.

Defect Management and Tracking

Defect Management and tracking is an important communication tool used by the tester. Defect tracking tools allow the tester to capture and report on defects to developers and the appropriate team members. Not only are tracking tools good for logging and tracking defects, but they also provide a large repertoire of value to the project and users. Defect management tools can link defects to business rules, test cases and regression scripts. They also allow the user to create custom reports to identify metrics such as defect trends and time-to-production.

Tools for this purpose exist that provide tracking and management capabilities while also keeping a repository of all defects recorded. One widely used tool is HP Quality Center. The main purpose of defect management is visibility and accountability. By the use of filters, users can easily see the defects they need to see without spending time sorting through non-pertinent information. Sometimes a business analyst may not have access to a defect management system and then will need to track and manage defects manually.

Whichever method is used, defect tracking is an important process for the tester. Defects are logged to record incorrect processes or other issues found. Subsequently, a defect report serves as a checklist of what needs to be fixed and re-tested. Defect tracking provides a good sense of the status of a project. Depending upon how far along a project is in the lifecycle, the number of defects found could be an indication of the actual level of completion of the project. As part of the risk assessment of each defect found, it may be determined that some defects are minor enough to wait to be fixed until after implementation to production so that the project is not delayed.

User Acceptance Testing (UAT)

The UAT sessions are critical to a project's success because they will allow the project team to assess whether their objectives have been met and/or whether further changes are necessary prior to implementation. The involvement of users in creating Test Plans/Checklists for UAT is critical to its success. Users are the stakeholders that have solid knowledge of what the system should do and the business purposes. They offer valuable insight into identifying test records, testing scenarios, and identifying issues with systems from the perspective of non-technical users. These users will use the system, possibly for the majority of their workday, and as such should be consulted on the following:

- Use of shortcuts and/or adaptability to Windows shortcuts.
- To ensure the functionality is visually ergonomic
- Navigational ease with multiple input capabilities, such as keyboard vs. mouse input.

When designing a Test Plan /Checklist for UAT, the designers of the Test Plan /Checklist not only need to consider the performance of any new functionality being added, but also how it may impact existing processing. It is advisable that the team working on the Test Plan /Checklist has one member to take the lead in organizing and

leading any meetings related to working on the documents, by serving as a Test Coordinator. The Test Coordinator should also be responsible for consolidating the group's work into one final draft. In addition, the Test Coordinator must keep the Project Manager and technical leads for the project apprised of the UAT group's progress and any issues that arise. The following testing types should be reviewed and incorporated as necessary when designing the test cases:

- Positive Testing – Any new “processing” scenarios are confirmed to work as expected
- Negative Testing – Any new “failure” or “error” scenarios are confirmed to work as expected
- Regression Testing - Existing functionality is not adversely impacted by changes
- Volume Testing - New/existing functionality can withstand a high volume of transaction processing.

Much like the format of the Test Plan used by the tester, the Test Plan /Checklist for UAT is a list of itemized test scenarios with a perspective of the user of the system. The document is not technical, and it should be written in everyday language. It is also written in a format where the successful outcome of the test is answered with a “Yes”. The UAT checklist has the following components:

- Numbering System
- System Being Tested
- Description of Transaction or Test Scenario
- Test Record
- Yes / No
- Test Result & Error Message (if result from previous column is “No”)

The users' time spent on UAT is time that is taken away from supporting their normal functions within their work unit, so it is crucial to use the time as efficiently as possible. With that in mind, the following should be taken into consideration:

- Have all testers confirmed availability to test according to the testing schedule?
- Do all testers clearly understand the objectives of the testing?
- Have all desired testing scenarios been identified?
- Have all necessary accounts/permissions for testers have been established and confirmed to be working prior to testing sessions?
- Do sufficient numbers of test records exist so that multiple testers can test at the same time?
- Is a clear communication plan in place and shared with testers to report issues with testing/test results?
- Is a defect management process in place to track and prioritize any defects found during UAT?

System Acceptance

System Acceptance is the final approval and sign off of the system testing and UAT. For documentation purposes, the sign-off should be written (such as an email), rather than verbal if possible. It is obtained by the person or people identified in the project documentation. Once the system has received signoff and approval, it is ready for implementation. Scheduling the implementation date should not wait until system acceptance. Rather, it should be contingent upon system acceptance.

Implementation and Training

Implementation

Communication and Coordination

===Migration Scheduling (Pre- and Post-Migration Testing===)

Training

User Experience

Usability vs. User Experience (UX)

Usability pertains to how easy a product is to use. The user is able to quickly and easily attain their goal. User experience (UX) includes usability, but also a number of other elements that help make the user's experience more enjoyable. Usability asks, 'Can the user accomplish their goal effectively?'. UX asks, 'Did the user have a delightful experience while attaining their goal?'.[1]

Peter Morville's honeycomb model illustrates what UX should encompass. In order for a product to have value, it should be useful, usable, findable, credible, accessible and desirable. [expand on each?]

In UX design, it is important to define priorities. Should a solution be more usable than findable, more accessible than credible? These decisions depend on the product being created.[2]

Why is UX important?

Why is UX important to New York State government? If we want people to like using our products, feel we are competent and credible, and have a desire to return to our products, then UX is important.[3]

Business Analysis and UX

Business analysis is concerned with business goals. User experience(UX) design is concerned with user goals. The two professions overlap a great deal, as illustrated in Berkley's BA-UX Continuum Model.

In an ideal situation, a UX designer might receive a rough product prototype from a BA, and then make improvements to it. Visual designers may also be involved to add interest and aesthetics to the prototype. The prototype could then be tested by and feedback attained from real users. However, because many companies/agencies do not have UX experts, the BA often plays the role of both.[4]

UX Best Practices

Implement a user centric approach. Define user profiles and personas during requirements elicitation. [4]
[more to come]

References

- ^ 1. User Interface Engineering - The Difference Between Usability and User Experience, retrieved 01/29/2014 from <http://www.uie.com/brainsparks/2007/03/16/the-difference-between-usability-and-user-experience/>
- ^ 2. Semantics Studio - User Experience Design, retrieved 01/29/2014 from <http://semanticstudios.com/publications/semantics/000029.php>
- ^ 3. Web Designer Studio - UI vs UX: what's the difference?, retrieved 01/29/2014 from <http://www.webdesignerdepot.com/2012/06/ui-vs-ux-whats-the-difference/>
- ^ 4&5. Business Analysis Times - Gaudi and Steve Jobs Way of User Interface Design, retrieved 01/29/2014 from <http://www.batimes.com/articles/gaudi-and-steve-jobs-way-of-user-interface-design.html>

References

- [1] http://en.wikibooks.org/wiki/Business_Analysis_Guidebook/User_Experience#endnote_UIE._2007
- [2] http://en.wikibooks.org/wiki/Business_Analysis_Guidebook/User_Experience#endnote_SS._2004
- [3] http://en.wikibooks.org/wiki/Business_Analysis_Guidebook/User_Experience#endnote_WDD._2012
- [4] http://en.wikibooks.org/wiki/Business_Analysis_Guidebook/User_Experience#endnote_BAT._2013

Business Process Improvement Models

Root Cause Analysis

What it is/Why Important

Root cause analysis, simply put, is a careful examination of a particular situation to discern the underlying reasons for a specific problem or variance. The Business Analysis Body Of Knowledge (BABOK) defines this as a “structured examination of the aspects of a situation to establish the root causes and resulting effects of the problem”. Depending upon the rigorousness of the examination conducted, it is possible to identify several layers of symptoms before reaching the underlying cause or causes of a particular situation.

When is it Used

Root causes analysis is most commonly affiliated with **Problem Solving**, although it can also be applied to organizational analysis, variance analysis, process improvement and software bug fixing. Essentially, whenever an outcome is less than ideal, it is generally possible to find a causal relationship or two or more. Given that some of the tools to discern root cause analysis can be subjective, it can often a judgment call as to the underlying contributing factor causing the variance—especially when the system undergoing evaluation is complex.

Questions to Consider

By carefully seeking out the root cause to a particular problem, and then applying some mitigation to the root cause, problems generally go away. By merely treating the symptoms of the problem, the underlying problem is likely to manifest itself in a new way, but not go away. Take for instance (dream up a good example). Often problems may not be severe enough to apply rigorous evaluation. In deciding how deep or quick to dive for the root causes, here are some questions to consider:

- What are the consequences of this issue/problem? Is it front page headlines? Life threatening or merely annoying?
- Is this a single occurrence or has it happened before?
- What is the probability of the situation occurring again?
- Were there events leading up to the problem/issue that could have served as an early warning signal?
- Was there a recent change prior to the occurrence which may have directly or indirectly facilitated its occurrence?
- Is this a system wide type of issue or is it limited to a single office or department?
- Are there controls in place to detect this type of issue/problem?

Sources of Problems

Root causes can be quite vast. Often it is a series of small problems and not just one single problem. The following list was adapted from Paul Wilson et al’s book “Root Cause Analysis” published in 1993. Having a list of contributing factors can often times help with identifying the actual root cause.

- Training (formal and informal)
 - Management Methods (resource and schedule planning)
 - Change Management (Modifications to existing process)
 - Communication (effective or not)
 - External (factors outside of the control of the agency)
 - Design (equipment and systems that support the work)
 - Work Practices (methods used to achieve the task)
 - Work Organization (organizing performance and sequence of tasks)
 - Physical Conditions (factors impacting performance)
 - Procurement (getting necessary resources)
-

- Documentation (instructions and procedures)
- Maintenance/Testing (including preventative maintenance)
- Man/Machine relationships (alarms and controls in place)

It is important to note that these potential root causes could be symptoms instead, and in some situations, it is possible to have multiple root causes.

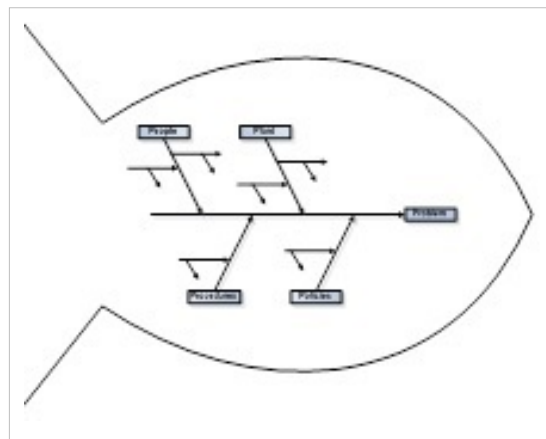
Methods

There are a number of tools that can be used in determining the root cause. Each of these methodologies will be explained briefly, a sample chart provided to illustrate the concept, and tools for construction and interpretation will be provided. The tools profiled include:

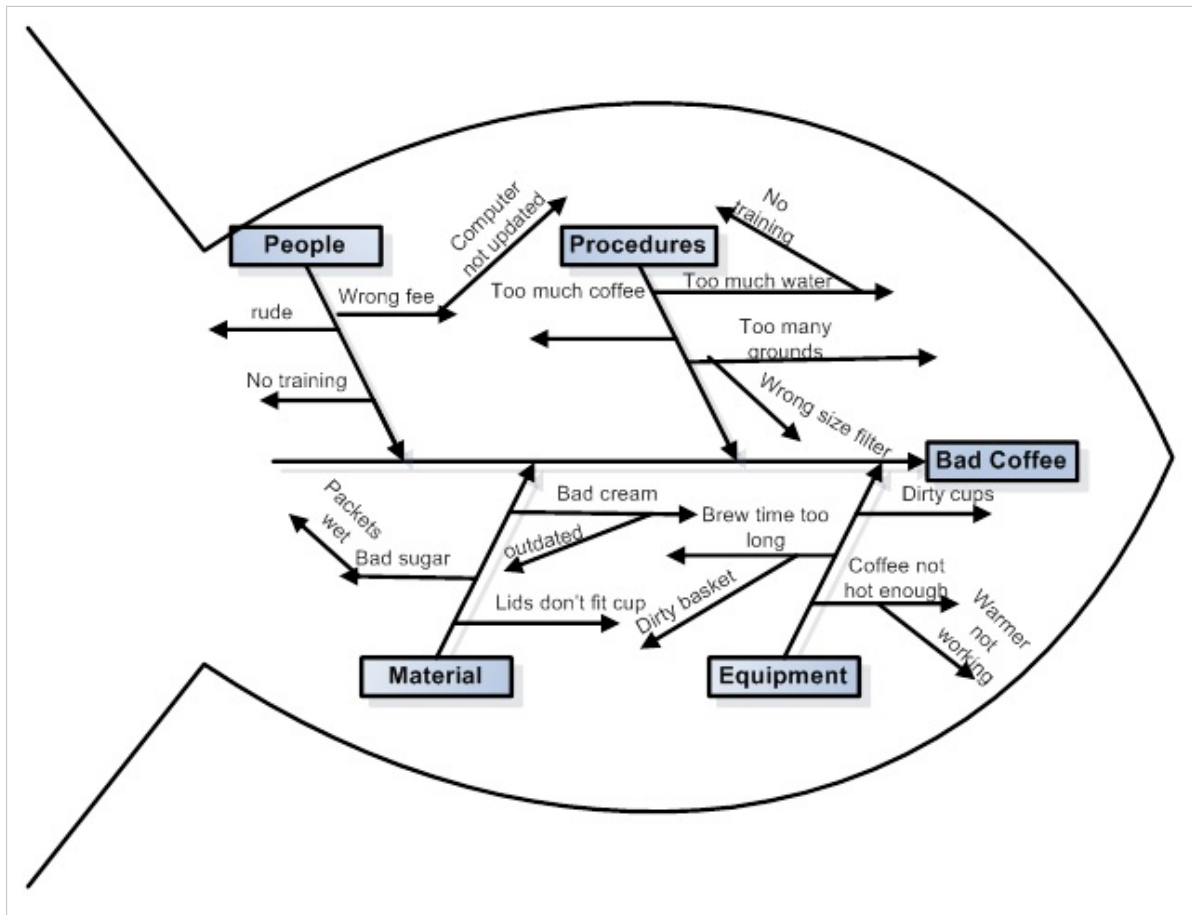
- Fishbone diagram
- Ask why 5 times
- Check Sheets/Pareto Chart
- Interrelationship Diagram
- RPR (Rapid Problem Resolution)

Fishbone Diagram

This tool is also referred to as the Ishikawa Diagram or the Cause and effect Diagram. It was named for Karou Ishikawa who pioneered TQM processes in the 1960s at the Kawasaki shipyards. It derives its common name from the shape of the diagram as evidenced in the Figure below:



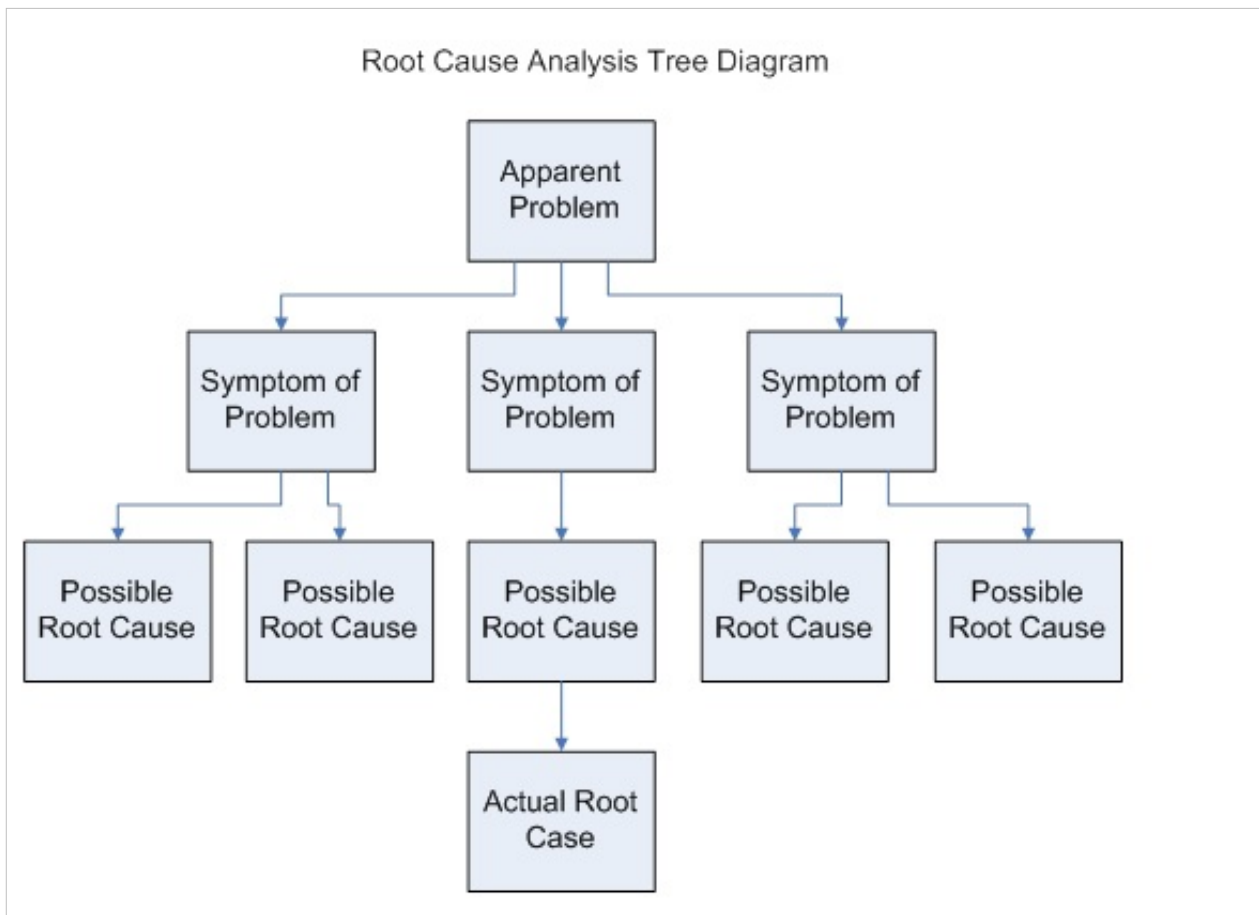
To construct a Fishbone Diagram, start with the problem you are addressing near the eye of the fish. From there, identify the primary causes of the problem. Typically, these are the 4 Ps or the 4 Ms, but can be what makes sense for the particular problem at hand. The 4 Ps are People, Procedure, Policy and Plant. The 4 Ms are Man, Machinery, Methods and Material. A sample chart showing an example of why a cup of coffee “could” be bad is as follows:



These diagrams can easily be constructed with pen and paper, and also various charting tools such as Visio (See business processes/cause and effect diagram). To analyze the results, look for common examples. Is something listed several times? In this instance, no training and poor quality inputs (e.g. bad sugar, dirty cups, etc.) appear to be very common themes to explore further. This is an excellent tool to use in a group setting.

Ask Why 5 Times

While it is easy to jump to a solution, it is often more difficult to pinpoint why something occurred. One of the most commonly used root cause analysis tools is referred to as the "5 Whys". This is based on the premise of continually asking why. Using the dirty coffee in the previous illustration, you could start with the apparent problem that the coffee is bad. By asking "why is the coffee bad", one of the first responses could be its weak. The next why would be, "Why is the Coffee weak", and the reply could be not enough coffee. In asking "why not enough coffee used", the reply could be we ran out. The asking of "why" continues until you get to possible root causes. To illustrate this concept, see the figure below:



A real life example on using this tool can be found in Washington DC at the Jefferson Memorial. The National Park Service noted that this monument was deteriorating at a faster rate than other DC monuments. By asking Why 5 times, they were able to get at the root of the problem as follows:

- Why is the memorial deteriorating faster? Because it was being washed more frequently.
- Why was the monument being washed more frequently? Because there were a lot of bird droppings.
- Why were there more bird droppings on the monument? Because birds were very attracted to the monument.
- Why were birds more attracted to the Jefferson memorial? Because of the number of fat spiders in and around the monument.
- Why are there a lot of spiders? Because of the number of insects that fly around the monument during evening hours.
- Why more insects? Because the monument's illumination attracted more insects.

In evaluating various solutions to this problem (e.g. pesticides, special coatings, different light, etc.), groups will identify different areas to focus on. In this particular case study, the Park Service chose to turn on the lights an hour later every evening. This one change reduced the bird dropping problem by 90%.

When using this technique, it is possible to follow different paths and derive different solutions. Should this occur, several factors can be considered when identifying the appropriate solution, such as what is within the group's ability to control. In the case of the Jefferson Memorial, they had the ability to control lighting and selected a no cost option that addressed the problem.

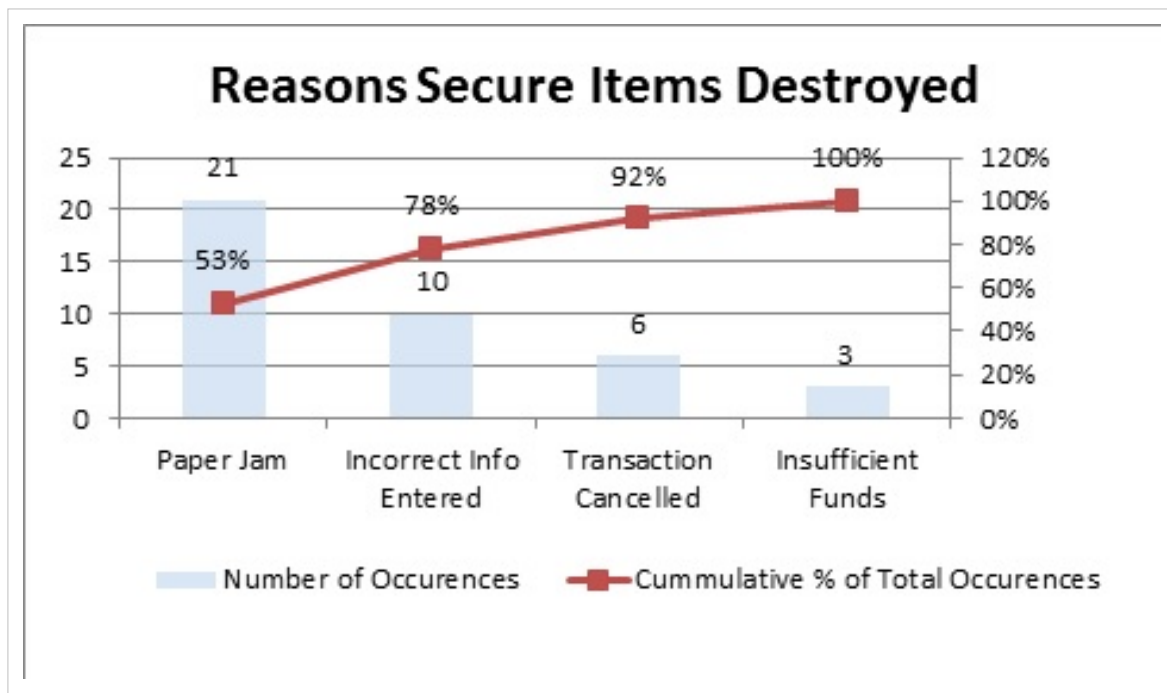
This technique is also used for requirements elicitation, particularly when interviewing subject matter experts. See the section Documenting and Managing Requirements.

Check Sheets/Pareto Chart

There is an old saying “what gets measured, gets done.” In the case of root cause analysis, the combination of creating a simple checksheet to collect data from observations or occurrences and charting onto a Pareto Chart can help pinpoint problem areas. In the absence of data, often perceived or apparent problems can lead you down the wrong path. By observing and recording the frequency of an occurrence for a specific period of time, it is possible to determine relative severity. See figure below for an example of a checksheet.

Cashier	Paper Jam	Incorrect Info Entered	Transaction Cancelled	Insufficient Funds	Total
A	4	2	2		9
B	10	5	3	2	20
C	3	2			5
D	3	1	1	1	6
Total	21	10	6	3	40

In constructing a check sheet, it's as simple as identifying the things you want to count and then counting as they occur. After a reasonable period of time, just count up the occurrences. In this example, the errors identified point to a paper jam (problem with paper and equipment) and incorrect information entered by the operator. For the paper jam—it could be the printer or it could be the material you are trying to print (weight, material, coating, etc.). To address this problem—it will be necessary to do several trial tests to help discern what the true root cause is. For the “incorrect info entered” it may be as simple as retraining cashier B or adding some behind the scenes edit checks to look for common errors. In all instances—it is best to focus on items within your immediate control and environment first, before trying to throw technology at the problem. Once the data has been collected, one powerful tool you can use to document the results is called a Pareto Diagram. The Pareto Chart displays the relative importance of problems or occurrences and is based on the principle that 80% of the problems result from 20% of the causes. The basis for the 80-20 rule was an Italian Economist, Vilfredo Pareto who noted that 80% of the land was owned by 20% of the people. By applying the results from the check sheet above, a sample diagram is below:



Note that the figure has two vertical axes. The one on the left provides a relative count of the number of occurrences, where the one on the right focuses on cumulative % of total occurrences. By focusing problem solving efforts on the

largest volume, the total errors will be reduced significantly.

Interrelationship Diagram

An interrelationship diagram is another valuable tool that helps to compare related issues in order to determine which ones are driving forces (root causes) and which ones are being influenced by others (symptoms). This exercise is best done in a group setting where you have a variety of perspectives. The matrices can take some time to get through, but typically provide valuable insights once completed. Using a list of symptoms/root causes, create a matrix (we are using a 5x5 example here), and then add 3 additional summary columns to the right.

For this example, we will look at causes of ineffective meetings and the 5 potential symptoms or root causes are: lack of an agenda, lack of facilitation, wrong people at the meeting, airtime dominated by a few and rehashing same stuff. For this example, the symptoms will vary by group and organization and not doubt with group input, it is possible to come up with more items. The small number is more to demonstrate how to construct, facilitate and evaluate the results. A completed matrix is below:

EDIT NOTE: Insert table representing interrelationship diagram for poor meetings

For each issue identified—ask the group the impact of each item against another. Starting with A. Lack of Agenda against B. Lack of Facilitation, ask the group, does A drive or influence B or does B influence A? Typically, if you have a facilitator, you often have an agenda so in this instance, an “up” arrow is placed on row A/column B to show impact of A on B and on row B column A. you will put an “in” arrow to show the influence of A onto B. Next you will look at Lack of an Agenda and Wrong people at the meeting. While you “could “ make a weak case that if you had an agenda, it could be obvious that you have the wrong people at the meeting—there are other drivers for this—so in this case—we will put in a “-“ dash signifying no relationship. For each pair—the matrix will receive a relationship mark. Once completed—it is time to add things up. All arrows pointing inwards (items being influenced) get added for each row and the sum is reported in the “In” column. All arrows pointing upwards (items influencing) get added for each row and the sum reported in the “Out” column and then both in and out are added together. In evaluating the results, look for the largest number of “out” as your root cause. In this example, the lack of a facilitator leads to rehashing the same stuff (meeting after meeting). This tool is also very good for determining critical processes, as well as root causes. Instead of listing problems or issues, record all of your processes with letters and evaluate which processes influence or directly impact other processes.

Rapid Problem Resolution (RPR)

This technique was designed specifically to identify the root cause of IT problems. While it is aligned with ITIL Problem Management Process, it requires that the problem be replicated and the method is designed to focus on a single symptom at a time until a root cause is identified. The method is comprised of three steps: 1) Discover 2) Investigate and 3) Fix. During the discovery phase, it is important to obtain as much information about the problem as possible (what is the problem, when does it occur, in what environment, frequency, etc.) and settle on what is the problem we are trying to solve. The investigate phase focuses on being able to replicate the problem so that it is possible to discern what is causing it. In this instance, it is necessary to develop and execute a diagnostic data capture tool so that results can be obtained to identify what is causing the problem. Once the root cause has been determined, then it is possible to trace where it occurs through reviewing diagnostic data. Once the problem is found, then a fix needs to be developed and implemented, and the solution verified.

In Summary

Root cause analysis is a critical component to problem solving. If you do not treat the root cause(s) of a problem, it is likely that the problem will not go away. By treating symptoms, the problem often manifests itself differently, offering a new set of symptoms. Since time and resources available to solve problems vary, it is good to have several tools available for seeking out root causes.

References

EDIT NOTE: Compile and add list following formatting rules

Strategic Analysis and Planning

Strategic vs. Tactical vs. Operational

The term strategic is often used interchangeably with tactical, which can lead to lack of clarity. Essentially, **strategic** is what you are planning to achieve and **tactical** is more of how you will achieve it. Examples of these two terms are as follows:

Strategy #1: Improve retention of key staff by 5%

Tactic #1: Create opportunities for key staff that appeal to them—allow time for R&D, implement their ideas, and allow them to participate on cross enterprise initiatives.

Strategy #2: Reduce maintenance costs by 10%

Tactic #2: Work towards eliminating legacy systems with high maintenance, seek to automate routine maintenance tasks, and establish controls to ensure work performed is necessary.

The term operational refers to day to day efforts to run the business. Once strategies and tactics are incorporated and achieved, they become how the business or organization operates.

Strategic Planning Components

Traditional strategic plans contain a variety of core elements to help the organization achieve success. Each of these serves a critical role in the overall plan development. At a macro level, they include:

1. Mission and Vision
2. Values and Behaviors
3. SWOT Analysis and Environmental Scans
4. Goals and Objectives
5. Measures
6. Strategic Projects

Each of these elements is detailed below and includes actual governmental examples.

Mission and Vision

The mission of an organization serves as a foundation of the organization and conveys what the organization does, who it serves and often describes what its primary purpose is. Noted examples found on the internet include:

- *NYS Department of Health*: We protect, improve and promote the health, productivity and well being of all New Yorkers.
- *The Office of Children and Family Services* serves New York's public by promoting the safety, permanency and well-being of our children, families and communities. We will achieve results by setting and enforcing policies, building partnerships, and funding and providing quality services.
- It is the mission of the *New York State Department of Transportation* to ensure our customers - those who live, work and travel in New York State -- have a safe, efficient, balanced and environmentally sound transportation system.

The vision is a statement which describes an organization's mental picture of success and where it is striving to be in 3-7 years. Without knowing where you are headed—then any path will do—but you may end up in circles along the way! A vision of what the future you are striving for can often help guide staff in the absence of specific directions.

Sample Vision Statements include:

- The *New York State Archives* provides unparalleled services that build, maintain, and provide access to New York's records to sustain a free, open, and democratic society and to support the cultural and intellectual life of all New Yorkers. We relentlessly pursue excellence in all our endeavors .
- *New York State Police* Vision: To build on our tradition of service.
- *NYS Department of Health* Vision: New Yorkers will be the healthiest people in the world - living in communities that promote health, protected from health threats, and having access to quality, evidence-based, cost-effective health services.

Values and Behaviors

One softer side of the strategic planning process is identification of values and behaviors that are critical to organization success. Research has shown that values typically influence behaviors, which in turn can drive results—particularly if the organizations culture and policies are aligned to support and recognize them. Organizations that adopt a standard set of values or behaviors need to ensure that the leadership of the organization demonstrates and role models what is acceptable. Clearly the values will be dependent on the type of organization. For example, an audit agency would likely value accuracy, credibility and integrity over innovative, customer service or respect for the individual, but perhaps an IT organization would value dependability, excellence, state-of-the-art and reliability. These values are often incorporated into mission statements or code of conducts.

Government examples in this area include:

Department of Motor Vehicle Operating Principles:

- We work with each other and our customers in an ethical, honest and respectful manner
- We protect the confidentiality and integrity of the information in our care
- We encourage the open exchange of ideas and perspectives and provide opportunities throughout the agency to suggest improvements

Department of Health Values: Dedication to the public good, Innovation, Excellence, Integrity, Teamwork, Efficiency

New York State Troopers Value:

- Integrity: To live and work in accordance with high ethical standards.
 - Respect: To treat people fairly while safeguarding their rights.
 - Customer Service: To ensure that everyone we meet receives dedicated and conscientious service.
-

- **Continuous Improvement and Learning:** To constantly improve ourselves and our organization.
- **Leadership:** To inspire, influence and support others in our organization and communities.

Enterprise Analysis

Enterprise Analysis, as described in the Business Analysis Body of Knowledge v2 is to “*assess the capabilities of the enterprise in order to understand the change needed to meet business needs and achieve strategic goals.*”

There are a variety of techniques that can be done to analyze the enterprise at a strategic level, and we will profile three of the most common ones here: environmental scan, SWOT analysis, and a customer and product/service matrix.

Environmental Scan: Prior to embarking on the development of goals and strategies, it is often valuable to conduct an environmental scan to see which, if any, appear to be influencing the organization more than others. By looking at the components of each factor and ranking them in order of importance, an organization can try to be more tactical in developing strategies to help achieve their goals.

- **Customers/Constituents:** Who are the people that are receiving goods and services from your organization? What are their requirements, how do they interface with you, what are their demographics?
- **Economy:** How is the current economy influencing your organization? What is the impact of inflation, unemployment, interest rates, credit availability and local taxes on the organization?
- **Human Resources:** How are we doing as an organization attracting and retaining top talent? Are our employees, manager and departments as productive as possible? Are we growing the right skills sets to enable us to thrive in the future?
- **Infrastructure:** This relates to the internal systems and operations of the organization and includes things like information systems, equipment, tools and procedures. Are they optimized or are we spending tons of time on maintenance and patching old equipment and systems?
- **Organizational Culture:** This relates to the values of the organization including norms, policies, rules and rewards. Are these aligned and support where the organization wants to be in the future?
- **Policy:** This variable relates to the overall objectives of the organization and includes policies, plans, targets, measures, key performance indicators and strategies. Are these aligned with the organizations values and goals?
- **Products/Services:** Do we have the appropriate mix of products and services that will help the organization meet client needs and the achievement of the organizational mission? Are our services offered in a manner that today's customer expects and demands?
- **Public Sector:** This references the influence of government such as laws, regulations and politics. Is government ensuring free markets and corporate growth, or are the policies stifling expansion?
- **Social:** Is the organization aligned and in tune with the overall needs of the citizens and the communities that they live in?
- **Technology:** This refers to the investment the organization makes in equipment, software, tools and telecommunications. Are we leveraging current technology and economies of scale, or have we diversified away our potential savings?

Once an organization sees themselves in the eyes of those whom they influence, then they are ready to conduct the next exercise, a SWOT analysis.

SWOT Analysis:

SWOT stands for Strengths, Weaknesses Opportunities, and Threats, and is a common technique used by organizations to develop a common understanding of where they are today. Using a simple 2x2 matrix, participants brainstorm the various areas as a starting point prior to development of strategies. A completed matrix is below for illustrative purposes:

Strengths: * Talented, diverse workforce. * Strong partnerships with industry associations. * Mature processes have been optimized to increase efficiencies.	Weaknesses: * Aging workforce. * Limited knowledge transfer tools.
Opportunities: * Need to grow social media presence. * IT transformation facilitates sharing of resources across agencies. * Optimize website to incorporate responsive design.	Threats: * Security, security, security. * Economy still rebounding. * Limited pool of funds for infrastructure investment.

Goals and Objectives

At a minimum, goals should include an action verb and a noun. They can be as succinct as Improve Employee Satisfaction, Provide Quality Customer Service or Protect Consumers. Typically they refer to where the agency wants to go, and the objectives are the actionable steps an agency employs to get there. One of the most widely used acronym to measure the effectiveness of goals (and requirements) is SMART, attributed to a paper by George T. Doran published in the November 1981 issue of Management Review called There's a S.M.A.R.T. way to write management's goals and objectives! In this instance, SMART stands for specific, measureable, assignable, realistic and time bound. For example, using the goal "Provide Quality Customer Service", possible key objectives can include:

- Increase/improve access to services through use of technology, including website enhancements
- Improve response time to customers
- Provide complete and accurate information to customers
- Provide fair and courteous customer service

Each of these is specific as a standalone goal. They convey a specific actionable action that can be taken that is not vague or left up to the reader for interpretation. These objectives could be measured—either through use of service delivery channels (mail vs. phone vs. internet vs. in person), cycle time process measures, and number of interactions needed to complete a transaction, and also customer satisfaction surveys. In terms of assignable—can someone take responsibility for the achievement and measurement of each of these goals? The next letter of the acronym is R for realistic. While you don't want really simple objectives, you should avoid totally unachievable ones either (e.g. world peace comes to mind)! Realistic and time bound go together. If you can take steps to move in a positive direction in achieving an objective, even if you cannot totally achieve it, then it is likely a worthwhile endeavor.

Measures

Often measures are referred to as key performance indicators. In recent years, with the proliferation of data and the tools needed to link disparate and/or mine large data sets, the ability to measure performance has been enhanced. For measures, there are a variety of different types, including process measures such as inputs and outputs, as well as outcome type measures aimed at effectiveness and efficiency of products and services delivered. Illustrative examples of each type are in the table below:

Type of Measure	Basic Types	Examples of Measure Type
Inputs refer to resources used to complete or improve operations	Resources Used	* level of personal services/man hours to perform a task * equipment usage (vs. idle time)
	Operating Costs	* warranty Cost * cost per transaction (by type of service delivery channel)
	Cash Expenditures	* % capital investment * % R&D
Outputs refer to the units of work completed	Completed Work	* % of project complete * number of workorders closed
Effectiveness refers to how well a product or service meets needs of the customer	Process Quality	* rework hours * number of visits/calls customer had to make to complete the transaction
	Satisfaction (customer)	* did products and services conform to requirements * eere employees courteous

	Satisfaction (employee)	* attrition rate * # of employee suggestions implemented
Efficiency refers to how effectively resources were used in delivering products and services	Timeliness	* project completed on time or early * process cycle time
	Operating Ratios	* supervisory span of control * direct to indirect costs

The key with performance measures, especially relative to strategic planning, is to select a variety of measures that show you how you are doing relative to your goals. When establishing measures, it is helpful to keep the following criteria in mind:

- The measures should be relevant to what you are trying to achieve
- It must be important to the stakeholders
- The data must be obtainable and cost effective to acquire and report
- It must be easily understandable by all who review it
- Must be used for constructive purposes.

One thing to keep in mind about data—people will typically give you the information they trust you to use. If people perceive the information will be used against them for non-constructive purposes, it is possible the numbers could be adjusted before receiving.

Strategic Projects

While measures and objectives are helpful in achieving goals, often an agency or organization needs to undertake a series of projects in order to make substantial progress towards achieving the goal. When organizations are evaluating their portfolio for project selection, one common criteria used is alignment with strategic goals. Will the outcomes and deliverables of the project help the organization to achieve one or more strategic goals? How closely is the project aligned with a goal—it shouldn't just "fit" but embody the essence of what the agency is attempting to achieve. If a project does not help foster a goal, and it is not critical to agency infrastructure, or offer a positive ROI, then perhaps it should not be undertaken. Projects deemed as "strategic" should be of a high enough priority that it gets the attention it deserves. Conversely, if every project in the portfolio is considered strategic and of a high priority, then you may have diluted the importance of all of the projects. See the section on prioritization for additional guidance on strategic project selection.

Session Planning Guidelines

When putting together a strategic planning session, there are several recommended guidelines for successful sessions, including:

- Allow enough time for thoughtful discussion. Perhaps give the questions in advance for key managers to solicit input from their staff beforehand. Typically strategic planning should be done over a full day, if not more.
- Ensure you have the right people. Strategic Planning requires leadership commitment to carry it out. If you do not have executive management on board, you will have difficulty implementing.
- Where possible, plan to do this off site. If people are close enough to the office, they will pop in and out during the day and not dedicate the time and effort.
- If possible, have smaller groups do some of the up front analysis and planning and have them present their results to the larger group for input and acceptance.
- Don't get so caught up in completing the technique that you forgot why you started the process
- Don't let the lack of data stop you from completing the process. Use your judgment and experience when needed.
- The plan is only as good as the follow through. If you don't work the plan, all you really have is a good doorstep at the end of it all.

Techniques to Engage Staff

Depending upon the importance and maturity level of planning within the organization, a variety of techniques can be employed to gain employee engagement. Some examples to consider are as follows:

- Seek input on mission/vision—ask people to provide examples of how they would apply it in their day to day job
- Ask employees to contribute their personal values up front and feed this into the plan
- Engage staff at the department/bureau/office level to do strategic planning, cascaded from the overall agency strategic plan
- Have employees develop process measures aimed at demonstrating achievement of the organizations goals
- Engage staff at all levels of the organization to participate in strategic planning in order to gain broader input
- Publish the organizations mission, vision and values statement on the intranet. Create posters of these and place them in the meeting rooms.

The important part is to communicate the plan to employees and them have them validate it. Does it ring true? Can they see themselves within the mission and vision?

Facilitating the Generation/Acceptance of the Plan

Once the planning has been completed, the statements drafted, the goals written and the measures established, it is time to put all the elements together in a document. The actual format or framework varies from organization to organization—but it should be something that is published and distributed throughout the organization—not only to your managers and employees, but also your customers, suppliers and stakeholders so they have a picture of what matters most and where you are going as an organization.

Ideally, once the plan is finalized, it is time to “work the plan.” This involves regularly measuring performance, and publishing the results so that all are aware how the organization is doing relative to its goals. Where possible, leaders should use the plan to help reinforce what matters—whether it is post it notes listing the mission, putting the current measure and target on a wall for all workers to see, a key phrase on a coffee cup, employee rewards and recognition noting excellence in the values/behavior area or sharing results upon completion of a strategic project. It is important to keep the plan alive through regular and frequent communication that reinforces key elements of the plan.

Resources

Local Government Management Guide to Strategic Planning, http://www.osc.state.ny.us/localgov/pubs/lmgm/strategic_planning.pdf

References

Doran, G. T. (1981). There's a S.M.A.R.T. way to write management's goals and objectives. *Management Review*, Volume 70, Issue 11(AMA FORUM), pp. 35–36.

LEAN

LEAN

Lean

Lean is a set of tools used by public, private and non-profit sectors to improve processes by removing waste and increasing efficiency. The core idea is to maximize customer value while minimizing waste and thereby creating more value for customers with fewer resources. The concepts have been successfully employed by manufacturing such as the Toyota Production System, but are equally applicable to all industries and services, including healthcare and government as they create a culture of continuous improvement.

The following is a five-step thought process for guiding the implementation of lean techniques:

1. Specify value from the standpoint of the end customer.
2. Identify all the steps in the value stream, eliminating wherever possible those steps that do not create value.
3. Make the value-creating steps occur in tight sequence so the product will flow smoothly toward the customer.
4. As flow is introduced, let customers pull value from the next upstream activity.
5. As value is specified, value streams are identified, wasted steps are removed, and flow and pull are introduced, begin the process again and continue it until a state of perfection is reached in which perfect value is created with no waste.

The following chart lists eight office examples of wastes that add costs to the business, but no value to the customer.

Waste Category	Office Examples ¹
Transportation – Movement of paperwork	Excessive email attachments, multiple hand-offs, multiple approvals
Inventory – Any form of batch processing	Filled in-boxes (electronic and paper), office supplies, sales literature, batch processing transactions and reports
Motion – Movement of people	Walking to/from copier, central filing, fax machine, other offices
Waiting	System downtime, system response time, approvals from others, information from customers
Overproduction – Producing more, sooner, or faster than is required by the next process	Printing paperwork out before it is really needed, purchasing items before they are needed, processing paperwork before the next person is ready for it
Over-processing	Re-entering data, extra copies, unnecessary or excessive reports, transactions, cost accounting, expediting, labor reporting, budget processes, travel expense reporting, month-end closing activities
Defects	Order entry errors, design errors and engineering change orders, invoice errors, employee turnover
Underutilized People – People's abilities, not their time	Limited employee authority and responsibility for basic tasks, management command and control, inadequate business tools available

1 Keyte, Beau and Locher, Drew. The Complete Lean Enterprise Value Stream Mapping for Administrative and Office Processes, 2004

Six Sigma

While the lean methodology concentrates on creating more value with less work, the Six Sigma system strives to identify and eliminate defects in product development. Many organizations combine the systems, Lean/Six Sigma, to improve both the method of production and the quality of the product. The six sigma quality management system is used to measure the number of defects that occur during a process. The system then determines how far this number deviates from a desired result. Six sigma is a set of methodologies used to achieve extremely low failure rates in any

process. The term six sigma derives from the mathematical use of sigma in statistics as a standard deviation. Six sigma is therefore six standard deviations which means less than 3.4 failures per million.

The main process for Six Sigma is DMAIC which stands for define, measure, analyze, improve, and control. Practitioners who have undergone extensive training in six sigma techniques and methodologies can be certified as six sigma green belts, black belts, and master black belts.

Lean for Government

In his book, *Extreme Government Makeover Increasing Our Capacity to Do More Good*, Ken Miller describes how government services have successfully applied lean concepts to improve services and cut non-value-added work. Rather than using the terms normally associated with lean, he makes an analogy to twisted water pipes impeding the flow of government services. He contends the pipes or processes employed by government are twisted because of excessive handoffs and reviews, backlogs, and other non-value-added work making the delivery of services slower than the demand from the public. His extreme government makeover techniques emulate the methods used on television to build a new house in a week without reducing the amount of value-added work or compromising quality. The steps, which are lean concepts and improvement techniques, include the following:

- **Expose and analyze the pipes** (value stream mapping). Determine where bottlenecks and unnecessary twists and turns exist in the pipeline.
 - **Poka Yoke the entry into the pipe.** Wherever possible, “mistake-proof” process components, i.e., change the process to make it easier to comply. Check lists are often used to make improvements in government compliance.
 - **Triage the flow.** Most government pipes are one-size-fits-all that every customer must travel through. Create separate pipes: self-service, everyday service and intensive-care service. Creating separate processes for self-service frees resources for those who need more assistance and the delivery of more service overall.
 - **Process simultaneously when possible.** Government tends to have long, sequential processes. With simultaneous processing, work remains the same but the duration is shorter. Find ways to move down-stream activities up-stream, thus shortening the pipe.
 - **Eliminate CYA by reducing handoffs.** Defense mechanisms are built to avoid getting blamed for others’ mistakes which takes time and management that is not directly related to producing the service. Reducing handoffs straightens the pipe. Trust workers to handle more complex tasks. Should government workers be treated as unskilled, uneducated, and unworthy of trust or as knowledgeable, skillful, and trustworthy?
 - **Cut batches.** Batch processing holds one customer hostage to a larger group and creates a dam mechanism into the pipe. Batches are convenient for the service producer, not for the customer.
 - **Break bottlenecks.** Identified as an area that does not have the capacity to handle all the work feeding into it so work piles up. Productivity depends totally on the capacity of the pipe and the system is only as good as its weakest link. Find ways to divert all but the most necessary work from the bottleneck.
 - **Eliminate backlog.** Cross-train staff so they can step in to help during high-traffic times. There are three kinds of backlog: Historic (not growing) – work overtime to dig out; Seasonal – get ready for it; Ongoing and growing – fix the process.
 - **Get off the crazy cycle.** Once work falls behind, it is destined to get far behind, due to status requests inundating the workload and status tracking taking time away from delivering the service. In order to stop the cycle the process must be made more efficient.
 - **Eliminate status requests.** Fewer status requests frees more resources for providing services and creates greater capacity.
-

Value Stream Mapping

Value stream mapping (VSM) refers to the activity of developing a visual representation of the flow of processes, from start to finish, involved in delivering a desired outcome, service, or product (a “value stream”) that is valued by customers. In the context of government, a value stream could be the process of conducting an audit, completing a procurement, or hiring new agency staff. VSM can increase understanding of actual decision-making processes and identify sources of non-value-added time (e.g., documents waiting to be reviewed). The typical products of a 2-5 day VSM workshop are a map of the “current state” of targeted processes and a “future state” map of the desired process flow and an associated implementation plan for future process improvement activities.

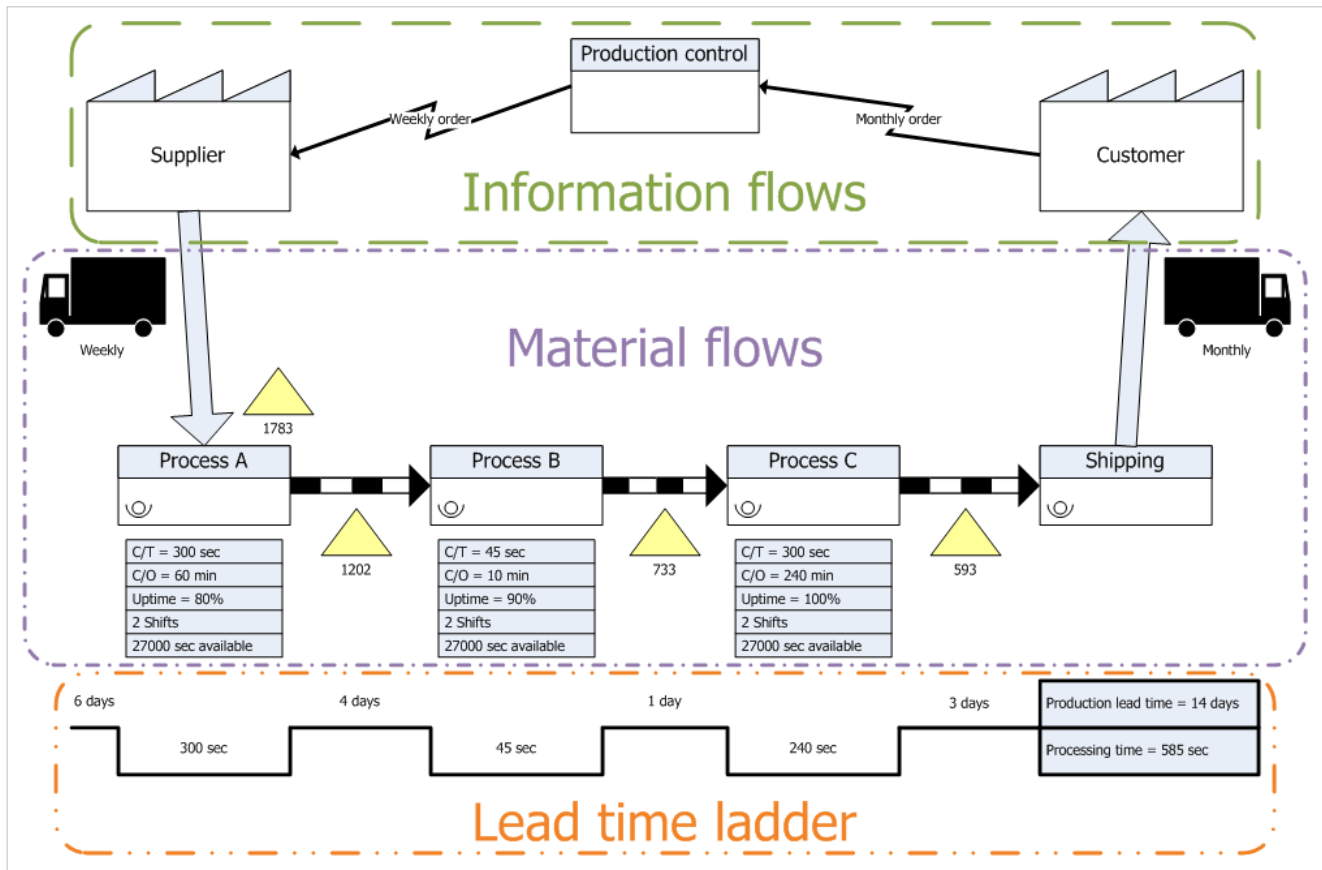


Image Source: WikiCommons Author: Daniel Penfield

For More Information

The following websites provide more information on Lean and Lean for Government:

www.lean.org ^[1] - Lean Enterprise Institute, Inc. – a nonprofit education, publishing, research, and conference organization

www.leangovcenter.com/govweb.htm ^[2] - LEAN Government Center Quality and Productivity Improvement Center – links to many city and state government lean initiatives

References

- [1] <http://www.lean.org>
- [2] <http://www.leangovcenter.com/govweb.htm>

Facilitation and Elicitation Techniques

Facilitation and Elicitation Techniques

Introduction

Requirements elicitation and facilitation skills are the cornerstone of the business analysis practice. Having accurate requirements is critical to effectively manage application development, business improvements or responses to current (changing) business conditions. As described in Section X of the guidebook, Business Analysts are responsible for facilitating discussion to gather, analyze and validate the requirements for a project and gain consensus on a solution.

Elicitation refers to the process of translating business needs into concrete, clear statements that can be managed and used to promote continuous improvement, ideally, across the entire operations of a business. Facilitation concerns the ability to extract true business requirements from the project customers, business stakeholders, project development team and/or external vendors. Business Analysts may perform daily tasks that are targeted to several areas of system development, business process improvement or business process reengineering, but must have a clear and correct understanding of the business needs captured in the requirements and the ability to manage them effectively to ensure the successful implementation of the project deliverable.

This section covers the general tasks that should occur to facilitate a successful productive meeting as well as commonly used techniques for business and technical requirements elicitation.

Facilitation Basics

Facilitated sessions provide structure and process to a group that is meeting to achieve a common goal. Facilitated sessions not only enhance communication among participants but encourage cross- functional participation and assist the group in making decision, solving problems or sharing ideas or information. In some cases, facilitated sessions can expedite the review and approval process. Within the business analysis discipline, facilitated sessions can be used for the following: defining business strategy, defining requirements, scope, business rules, and process requirements, perform walk-throughs and reviews. A successful meeting is dependent on three key areas: planning, conducting the meeting and follow-up.

Planning

Significant time should be spent on the preparation of a meeting. When planning a meeting, three key questions should be asked:

- Why is the session being held (objective/purpose)?
- What specific outcome is required from the meeting (deliverables)?
- What key participants or decision making authority should attend the meeting (participant list)?

One of the first tasks the facilitator is responsible for is clearly defining the purpose and objective of the meeting. The facilitator may discuss with the Project Manager and/or Project Sponsor, if necessary, the scope, objectives and specific outcomes that are required from the meeting. While planning for the session, consideration should be given to ensuring the necessary participants are invited to the meeting. Requirement gathering sessions will require business and technical subject matter experts. If the purpose of the meeting is to reach a decision on a proposed

solution, participants with decision making authority should be invited to the meeting. In addition to the role of the facilitator, other roles can be assigned at a meeting. These roles are described below.

- **Facilitator-** The facilitator is responsible for identifying those issues that can be solved as part of the meeting and those which need to be assigned at the end of the meeting for follow-up investigation and resolution. A facilitator helps understand a common objective to achieve a goal without taking a side in discussion and encourages discussion and idea generation and possibilities by engaging stakeholders.
- **Scribe-** an individual who is assigned to document the minutes of the meeting. The scribe may be asked at the end of the meeting to recap the action items for the group.
- **Timekeeper-** an individual who is assigned to monitor time to ensure all agenda topics are covered at the meeting.
- **Participant or Subject Matter Expert (SME)-** Participants attend and provides input into the sessions. Subject Matter Experts are generally the closest to the subject matter. Not having the right SMEs in the meeting will become a risk in trying to achieve the meeting outcomes.

The meeting agenda is the roadmap that structures a meeting around the group's purpose and helps to keep the meeting on track to achieve the needed outcomes within the time available. Although different formats for agendas exist, an agenda should include the items listed below. Some meeting agenda templates may also provide space for open action items and/or issues, recorded outcomes and next meeting date/time. Items to include in an agenda include:

- Meeting identification, including the following items:
 - Name of Meeting
 - Date of Meeting
 - Time of Meeting
 - Location of Meeting
 - Invited Attendees
- Objective of Meeting
- Who will be involved in presenting a topic
- Estimated time to cover a topic
- Any separate documents that will be reviewed

When developing the agenda, consideration should be given to structuring the agenda so the meeting starts with less controversial items, building towards more controversial items in the middle of the meeting and ending with items about which you can anticipate agreement. Meeting agendas and associated materials should be sent out at least one to two business days before the meeting.

Conducting the meeting

At the start of the meeting, the facilitator should review the agenda and the objective of the meeting. It is important to make sure that the participants have a clear understanding of what specifically needs to be accomplished at the meeting.

Depending on the size and purpose of the meeting, ground rules may also need to be established and reviewed prior to discussing the agenda topics. Ground rules establish boundaries and help create an environment where individuals feel comfortable participating in a meaningful way. Depending on the purpose of the meeting, appropriate ground rules may include:

- Limit discussion to x minutes per item
 - Lengthy issues should be documented and tabled
 - One person speaks at a time
 - Avoid side discussions
 - The whole group is responsible for the results
 - One person speaks at a time
-

- Allow individuals to finish their idea/thought
- Criticize the product or process, not people

It is the role of the facilitator to bring structure to a meeting and lead individuals through the agenda to reach consensus on a decision or elicit requirements. For a session to be successful, the facilitator must redirect the group when necessary, engage in active listening, generate participation from all participants, paraphrase and question to expand dialog and document information on flip charts. In addition, it is important that the facilitator remain neutral in order for participants to feel open to generate and discuss ideas.

At the close of the meeting, the facilitator should identify next steps and solutions that have been selected or decisions that have been made. If time warrants, the facilitator may also solicit any further feedback or comments.

Follow Up

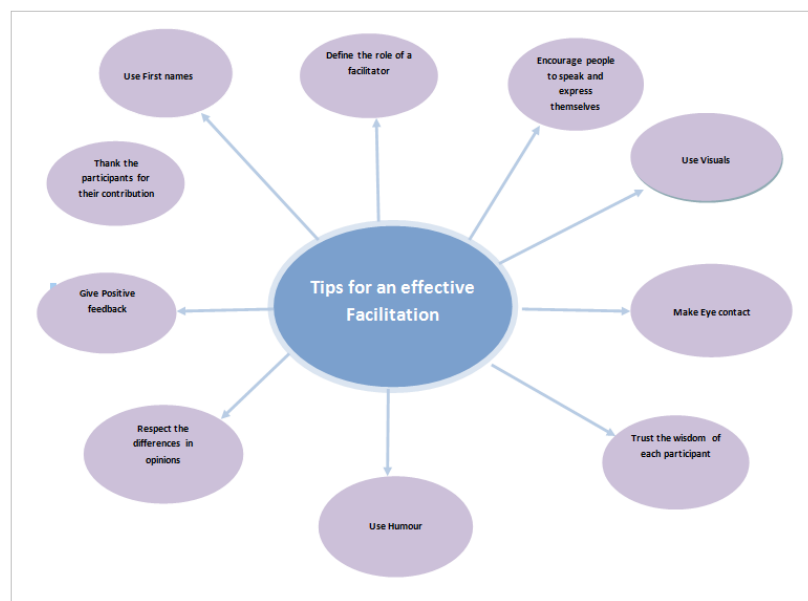
After the meeting is conducted, the facilitator or participant(s) should follow up on open issues. The designated scribe distributes the meeting minutes to all the invited participants. Clarification and/or follow-up may be received from meeting participants. Participants should provide statuses on any action items given at the meeting. Also if necessary, the facilitator schedules additional meetings.

Facilitation Techniques

The role of the facilitator is to foster and encourage discussion and depending on the meeting purpose, idea generation. Below are some tips that a facilitator can use to structure the discussion:

- Group or Individual Brainstorming
- Ask open ended questions to generate ideas
- Active listening and Paraphrasing
- Encourage equal group participation
- Ask for input
- Use Flip Charts to document information

As a business analyst, it is not uncommon when the analyst must not only facilitate the meeting but also contribute to the discussion to gather requirements, solution options or reach consensus on a recommendation. The chart on the next page includes several types of questions that can be used to help elicit information from stakeholders.



EDIT NOTE: Insert table of facilitation techniques from skillport

Troubleshooting and Dealing with Difficult Behaviors

There may be times during a meeting, a meeting participant presents difficult behavior or conflict arises between two meeting participants. In this event, it is the facilitator's role to manage the conflict. The table below highlights the most common difficult behaviors and a recommended approach that the facilitator may take to correct the behavior to keep the meeting on track.

Difficult Behavior	Approach
Non-stop Talking	Summarize the points made and then call upon else to continue the discussion.
Displaying Superiority	Recognize the participant's contribution and ability and ask them the more challenging questions.
Repeating the Same Points	Reassure the participant that you heard and recorded his/her point. Ask them if they have an additional point they would like to make.
Side Conversations	Tell the person that you didn't hear his/her comments and ask him/her to repeat them for the group. Ask the participants who are having the side bar conversation if there is anything they would like to add.
Anger	Attempt to translate their feelings into specifics that are within the power of the group to deal with.
Doubting	Reassure the participant that the group will carefully evaluate and judge the viability of all ideas at a later point in the process.
Monopolizing the conversation	Avoid eye contact with the participant and select others within the group to offer their thoughts.

Common Requirements Elicitation Techniques

Many techniques are available for business or system requirements elicitation. This section describes the commonly used techniques. Depending on the size and scale of the project, several of these techniques may be combined to ensure a complete picture of the requirements has been achieved. The techniques below are grouped into three categories: Vision Development, Analysis, Definition and Other. Techniques used to develop a vision or used to generate ideas for new solutions or a proposed approach. Analysis techniques are best used to conduct a gap analysis, perhaps when comparing a current environment to the envisioned target environment. The facilitator should select the most appropriate technique based on the meeting objective. The section is not exhaustive of the many techniques that a facilitator can employ.

Vision Development Techniques

Brainstorming

Brainstorming is an effective technique for identifying a diverse group of ideas, a new or alternative solution, or a vision within a relatively short period of time. Brainstorming works by focusing on a topic or problem and helps answers questions like:

1. . What options (or alternative options) are available to resolve problem?
2. . What factors are contributing to not moving forward with an option?
3. . What are possible causes for a delay in product x?
4. . What are possible solutions to problem x?

Brainstorming sessions allows participants to come together and creatively think of what a solution may look. At the beginning of a brainstorming session, participants should be reminded of the simple ground rule that no idea is a bad idea. The best ideas are often generated when individuals are creative and build on the ideas of others. As ideas are generated, the facilitator should document them on flipcharts or sticky notes for participants to review. Either at the conclusion of the meeting or after the session, the ideas are consolidated and/or duplicates eliminated. A consensus is reached as to what solution is best. Finally, after the meeting the outcomes should be distributed to meeting participants.

Several options exist for the structure of a brainstorming session. Below are a few techniques that may be used depending on the scale and scope of the meeting.

- Open Discussion- free flowing, no particular format. The lack of structure to this type of discussion requires skilled facilitation.
- Informal Group Brainstorming- anything goes, no critiquing of ideas, build on ideas of others, all group members call out ideas in no set order. Ideas recorded on flipchart for review, consolidation and decision making.
- Formal Group Brainstorming- seeks input in a predetermined order asking each member to share one idea at a time. An individual may pass. Continue to work through the group until all ideas are presented. This structure ensures that all participants have the opportunity to participate.
- Group Brainstorm on Post-its- each person records one idea per post-it note, calls out in no particular order. The use of post-it notes allows ideas to be moved around easily for consolidation, discussion and elimination. In addition, having participants write information on post-its also keeps them engaged in the meeting.
- Individual Brainstorm on Post-its- each person works individually to record an idea per post-it and ideas are shared in a predetermined order. This structure ensures that everyone participates.
- Individual Brainstorm and Report out- each person works individual to record an idea. One individual is selected to share entire list. Other participants report any additional ideas.
- Mind Mapping- print focus idea in a circle or box in center of page, print key ideas or thoughts on lines connected to the center focus, build from key ideas all related ideas using branches from the key ideas. Show connectors and groupings to various branches of the map.
- Partners- have people work in pairs to discuss an issue and brainstorm their ideas

Brainwriting

Brainwriting is a similar technique to brainstorming. The main difference is that brainwriting is anonymous. Individuals participating in the session write their ideas down and share with the group to further brainstorm the idea.

Definition Techniques

Focus Groups

Focus groups may be used to gather input into design or feedback from individuals who are directly involved with a process. Focus groups are held with customers, subject matter experts or end users to discuss a process or technology and share their perspectives. Focus groups are a good technique to learn about opportunities for improvement, and customer needs and problems. Although not as common, focus groups can also be used to gather and document requirements.

Joint Application Development

Joint Application Development (JAD) is a commonly used technique to gather requirements. JAD sessions can be used in a variety of other purposes in the system development, business process management and project management lifecycles. They can be used to generate ideas for new system features, review and agree to specifications of a system or gain consensus on the objectives of a project.

JAD sessions are useful to elicit large amounts of information within a relatively short time frame. They are typically a one to two day focused session allows stakeholders to come together in a structured setting. Because of the amount of information that a business analyst is able to obtain from one JAD session, they generally accelerate systems development. JAD sessions are also most effective when participants are able to make decisions regarding the system or process. <http://www.ksinc.com/itpmcptools/JADGuidelines.pdf>

Analysis Techniques

GAP Analysis

GAP analysis is an effective technique to compare differences between two different environments or systems. In business analysis, gap analysis can be used to determine the difference between the business requirements and system capabilities or studying two different environments (current and target environment). The information is then used to determine what is needed in order to move from one state to the other. Gap Analysis is also a useful technique to capture missing or incorrect system requirements. Although several templates exist for capturing information obtained in a gap analysis exercise, below is a simple template that can be customized to fit your needs.

Current State	Future State	Next Steps
-	-	-

Other Techniques

Surveys/Questionnaires

Surveys and questionnaires are an informal means to elicit requirements. Surveys are useful to reach a large audience. Surveys can be designed to characterize requirements or envisioned components of a system. Surveys should be short, to ensure a higher response rate. Development of surveys requires preparation to ensure the needed information will be obtained from the questions asked. Survey questions may be developed as open or closed questions that require.

Observation/shadowing

Observing a user is an effective technique when gathering requirements pertaining to an existing, current process. If an end user has been in their current position for many years, they may have a difficult time describing the processes they routinely follow. Observing someone doing their job will provide the Business Analyst insight on the overall process as well as bottlenecks. Observation/shadowing allow the business analyst to uncover requirements that may have been easily overlooked. The technique is effective for analysis of workflow process modeling or business process reengineering.

Interviews

Interviewing stakeholders either in an individual or group setting is a straightforward approach to obtaining requirements. Interviewing stakeholders or end users allows the business analyst to ask open ended and probing questions to uncover the necessary requirements and understand their expectations. At the end of the interview, the business analyst should send their notes to the interviewee (or group) for review.

Document Analysis

Reviewing existing (as-is) documentation for an existing process or system can be useful, if available. Evaluating this documentation can help the business analyst complete a gap analysis in order to ????

Procedures, requirements documents, proposals. After reading and understanding the current process or system, the BA may document questions and follow-up with a SME. Use existing documentation to uncover new requirements.

Models

One additional facilitation technique for Business Analyst's is the development of models. Workflow models and/or data models can provide business rules or the sequencing of work that will aid in discussion. Please see section X in this Guidebook for additional information and benefits of modeling.

Prioritization Techniques

“Success is only another form of failure if we forget what our priorities should be.” – Harry Lloyd

Prioritization and When to Use

Prioritization focuses on what is most important—whether it be project deliverables or an organization's goals—prioritization helps to rank the multiples into what matters most. We are all faced with the need for prioritization, whether it's juggling a large number of tasks, the email flag noting a high priority or in a medical emergency room, where those with the greatest need are served first. The ability to prioritize can serve organizations and their staff well—when faced with multiple demands within a finite timeframe, a relative ranking of what is more important than something else can guide employees to work on the right things.

Typically, a good rule of thumb for prioritization is when you have more than three items, issues, requirements, projects or goals—discerning the relative importance of the items against one another offers value to those who must use them.

There are a number of different techniques that help organizations and teams rank the relative order of importance of like items, including:

- Strategic Alignment
- High-Medium-Low or Small-Medium-Large
- Weighted/Criteria
- Ranking
- Matrix
- Impact Analysis

Each of these are detailed below with examples in order to provide guidance in their completion.

Prioritization Techniques

While many different techniques are listed below—they are often paired together. For example, strategic alignment may be one criteria in a weighted matrix! Each are presented separately in recognition that they can be done individually.

Strategic Alignment

Strategic alignment refers to the linkage between what the organizations is trying to achieve, e.g. vision, goals, and values against a core set of deliverables, such as projects, priorities, or business processes. Alignment refers to comparing each of the deliverables against what the organization is trying to achieve to discern relative fit. If items are strongly aligned, then it is surmised that there is alignment. When items kind of fit or do not seem to match up—then one could surmise there is not an alignment. In most instances, these are a judgment call. Items which appear to offer real value in meeting the organizations vision, goals and values should receive a higher prioritization. Those deliverables that meet more than one strategic area should be rated even higher than those that link to a single element.

High-Medium-Low

In its simplest form, High-Medium-Low is subjective criteria that could be applied to a list of deliverables that use relative size as the criteria. While size is used most often when ranking items high-medium-low or small-medium-large, the same criteria can also be applied to complexity, duration, benefits and cost relatively easy.

Caution! When applying the use of small-medium-large or high-medium-low to multiple criteria—it is critical to ensure that when applied, all are referencing the same “desired” outcome. Merely assuming “high” is good in all categories may result in ambiguity in your results. For example, if we apply benefit and cost using high as a good outcome, we will tend to favor expensive outcomes (cost=high=good) over other options or alternatives. This caution will be detailed further later in this section. Use of high-medium-low as a simple prioritization technique is detailed below:

Options	RISK Low-Good, Med-Depends, High-Bad	COST Low-Good, Med-Depends, High-Bad	TIME TO DO Low-Good, Med-Depends, High-Bad	RANK
Option A	High	High	High	3
Option B	Med	Med	High	2
Option C	Low	Low	Med	1

As noted in the above example, items ranked low are preferred over items ranked high.

Weighted Criteria

This form of prioritization recognizes that all criteria are not created equal. A weighted criteria approach allows the prioritizers to identify which criteria are more important and assign a higher value/weight to the criteria. An example of this form of prioritization can be found in the decision making section.

Weighted criteria is generally accepted as the best form of prioritization, however it does require additional steps and is viewed as taking too much time or as too complex in some organizations.

Ranking

This form of prioritization is seen as the most difficult to undertake when done alone. The more items to rank, the more difficult it is to classify. Given the subjective nature of the analysis, e.g. item “C” is more important than “A” and “E” is more important than “C”, etc., this is often best done in group settings with 2-3 people assigning the relative importance.

There are several techniques used to help facilitating rank ordering items, namely a clear focus as to why they are ranked as they are and using pre-sorting to ease completion. The clear focus technique identifies a single set of values by which to make the decision of what is most important. Often in business, the focus is a positive impact on the bottom line. Conversely, in government and non-profits will often look at what is best for citizens/clients. The higher likelihood of a beneficial outcome, the higher the item will be ranked.

Using a pre-sort technique suggests sorting items into different categories before ranking, such as looking at items from a high-medium and low perspective. The items with high benefits are put in the top category and those with lessor benefits are put in the bottom category. Once all items are initially sorted, ranking within the individual categories is an easier task. Once completed, it is best to revisit the entire list to ensure that it is ranked appropriately and that each rank number (1,2,3,4) is used only once.

Matrix

Matrices can come in various shapes and forms—with the most commonly done in squares or linear fashion. Square matrices typically have the criteria on the top and the alternatives being ranked in the left hand column as shown below:

Option	Criteria A	Criteria B	Criteria C	Rank
Option 1				
Option 2				
Option 3				
Option 4				

Another form of matrix is a linear one, often referred to as a quadrant, which uses a 2 dimensional Cartesian system and typically two criteria. While seemingly more complex, it offers a unique visual representation of alternatives that can facilitate decision making.

To illustrate this tool, the following example and resulting chart will evaluate which projects a particular organization should undertake.

INSERT PEARL/etc. file...send from work.

This type of matrix has many alternate uses. For example, Gartner, a major IT consulting firm uses a similar matrix to profile multiple vendors in a similar space and call them magic quadrants. Another application is called a customer window where you apply what a customer wants versus what they receive they are getting, as a form of gap analysis.

NOTE: Need to finish this section

BA Software Tools

BA Software Tools

Business Process Management (BPM) and Diagramming Tools

Requirements Management Tools

Drawing Tools

Communication Skills

Communication Skills

Introduction

By nature of the job, business analysts spend a great deal of time interacting with users, clients, management and developers. A project's success depends upon the business analyst clearly communicating details like project requirements, requested changes and testing results. Hence for a business analyst articulate language skills and outstanding written communication abilities are the key to success. In case there are disagreements or conflicts, effective communication can be again used for solving such issues.

Communication Types:

Understanding the communication type, will help in having a better understanding of communication skills. There are many classifications of communication, based on communication channels, purpose and group size.

Types of communication based on the communication channels used:

Verbal Communication:

- Oral Communication
- Written Communication

Nonverbal Communication:

- Appearance
- Speaker: clothing, hairstyle, neatness, use of cosmetics
- Surrounding: room size, lighting, decorations, furnishings
- Body Language: facial expressions, gestures, postures
- Sounds

Types of Communication based on Purpose and Style:

- Formal Communication: Presentations, documents, discussions, surveys
- Informal Communication: Notes, discussions, conversations

Types of Communication based on the size of the audience:

- Interpersonal communication
- Intrapersonal communication
- Small-group communication
- Public communication

Communication Skills

No matter what the communication type is, there are few general communication principles business analyst needs to keep in mind. A few things to keep your eyes on while practicing the fine art of communication are:

1. Make the message easy to comprehend Simplify, simplify, simplify. Communicate what you need to say in as few words as possible. Some of your readers or listeners may be able to understand a complex software textbook but fact is every one of them can understand a newspaper head line, so try to eliminate jargons and abbreviations

2. Tailor the message to the audience BABOK states that a sign of strong writing skills is the "ability to adjust the style of writing for the needs of the audience." Software developers may not need to be sold on the awesomeness of the new product, but a customer who's providing early feedback or is part of a virtual focus group will.

3. Active Listening Although people think that they are listening when another person talks, actually they are spending time planning what to say next. It is important to listen to what the other person says and then come up with what you want to say. Also Repeating back what you understand the speaker to be saying, and verbalizing what you interpret from that gives the speaker a chance to clarify, and eliminates assumptions.

4. Stay positive and friendly If you're tired or frustrated, try not to show it. It will make people less inclined to help you, or listen to you. For every audience, stay positive—whether presenting a problem or a solution.

5. Stay Focused Very often a discussion or an argument strays away from the topic, so it is important for a business analyst to remain focused to keep the stakeholders in track of the end goal

6. Understanding Others Point of Views In most of the communications, we want ourselves heard and understood. We talk a lot on our point of view and try to get the buying of who are listening. If you want them to hear you, you need to hear them and understand their point of view too.

7. Empathy When Criticizing It is really important to listen to the other person's pain and difficulties and respond with empathy.

8. Taking Ownership Taking personal responsibility is strength. When it comes to effective communication, admitting what you did wrong is respected and required. This behavior shows maturity and sets an example.

9. Compromise if Necessary Communication is not about winning. It's about getting things done. For the objective of getting things done, you may have to compromise in the process.

10. Take a Time-Out if Necessary Sometimes, you need to take a break in the middle of the discussion. If the communication is intensive, there can be ineffective communication pattern surfaced. Once you notice such patterns, you need to take a break and then continue.

11. Ask for Help Sometimes, you might have difficulties communicate certain things to certain parties. This could be due to an issue related respect or something else. In such cases, seek help from others. Your manager will be one of the best persons to help you with.

Acquiring Communicating Skills

Soft skills like communication skills are not easy to learn and are acquired overtime. But being conscious about them will slowly take you to the level you want to be in communication. If we develop a process to communication skills, we can build a methodology for ourselves. And if we have a methodology, we can master it fairly quick. So let's now look at one of the process model/tool, which is commonly accepted in the sociology field. This process is based on the sender /receiver concept. Communication is the process whereby we attempt to transmit our thoughts, ideas, wishes, or emotions to a specific audience. The goal of communication is the acceptance of the sender's message by the receiver.

S.M.C.R. - SENDER-MESSAGE-CHANNEL-RECEIVER Model

Sender The sender (or source in the S.M.C.R. model) is the transmitter of the message. There are four factors which influence the sender in any communication he/she transmits:

1. Communications skills
2. Attitudes
3. Knowledge
4. Culture

Communication skills: There are five verbal communication skills which determine our ability to transmit and receive messages. Two are sending skills: speaking and writing. Two are receiving skills: listening and reading. The fifth is important to both sending and receiving: thought or reasoning. The extent of the development of these skills helps determine our ability to communicate verbally. The effectiveness of our communications is also determined by our ability with nonverbal communications skills. A stern look of disapproval from the group leader readily communicates to the group member receiving the look that something he/she said or did was not well taken.

Attitude Attitude influence our communication in three ways:

- Attitude toward ourselves: If we have a favorable self-attitude, the receivers will note our self-confidence. If we have an unfavorable self-attitude, the receivers will note our uneasiness. However, if our favorable self-attitude is too strong, we tend to become brash and overbearing, and our communication loses much of its effect with the receiver.
- Attitude toward subject matter: if a business analyst does not approve for a requirement he/she will miss to see the positives it will bring to the project
- Attitude toward/from the receiver: Our messages are likely to be very different when communicating the same content to someone we like, to someone we dislike, to someone in a higher position than ours, in the same position, or in a lower position.

Knowledge Subject matter knowledge has a bearing on our ability to communicate effectively about a subject. It is important for a business analyst to have domain knowledge.

Culture Our culture also influences our communication style. It is important for a business analyst to be aware and sensitive about audience cultural background.

Message In the S.M.C.R. Model, the message is what the sender attempts to transmit to his/her specified receivers. Every message has at least two major aspects:

1. Content
2. Treatment

The content of the message includes the assertions, arguments, appeals, and themes that the sender transmits to the receivers. For instance, for business analyst message content may contain arguments in favor of a decision supported by feasibility analysis and results of the survey.

The treatment of the message is the arrangement or ordering of the content by the sender. The receiver is likely to be more receptive to the message, if the sender talks about the supporting documents prior to presenting the decision.

Channel Social Scientists recognize two types of channels:

1. Sensory channels
2. Institutionalized channels

Sensory channels are based on the five senses of sight, sound, touch, smell, and taste. Institutionalized channels include face-to-face conversation, printed materials, and the electronic media.

We use the institutionalized means to transmit most of our messages. Each institutionalized medium requires one or more of the sensory channels to carry the message from the sender to the receiver. For instance, when we use face-to-face conversation, we make use of sight (gestures, expressions) and sound (voice tones)

Social Scientists have generally found that the receiver's attention is more likely to be gained if the sender uses a combination of institutionalized means using two or more sensory channels. For example, if you make a prototype of the proposed solution along with description the stakeholders will find it easy to make a decision

Receivers The receiver in the S.M.C.R. model must attend to, interpret, and respond to the transmitted message. The goal of communication is reached when the receiver accepts the sender's message. For being a good receiver, you need to have two means:

1. Attention
2. Comprehension

Receivers are more inclined to accept messages which are sent by sender using the above mentioned techniques

Feedback Feedback is the sender's way of determining the effectiveness of his/her message. During feedback the direction of the communication process is reversed. Here the original receiver goes through the same process as did the original sender and the same factors influence him as they did the sender. Feedback provides a method of eliminating miss-communication. It is most effective in face-to-face conversation where feedback is instantaneous.

Feedbacks ensure that different tasks and processes remain aligned with the project goal. Also a BA acquires business approval and sign-off from stakeholders via feedbacks.

With these sets of communication principles and tool in your pocket, as a business analyst you are ready to develop a communication plan for any project which outlines what must be communicated, to which stakeholders, at what time and in what format. Happy sharing!

Building Relationships and Trust

Why Trust is Important in Organizations

Analytical Thinking and Problem Solving

Analytical Thinking and Problem Solving

Underlying Competencies

Decision Making

EDIT NOTE: Add graphic for Decision Making

Arriving to decisions can be a daunting task because it involves many factors. One needs to understand that a high quality decision comes with a warrant: Not a guarantee of a certain outcome. But with good decision making tools we have a warranty that the process we used to arrive at a choice was a good one. As a Business Analyst you need to make sure that you take a systematic approach towards decision making. Some guidelines are:

- **Create a Constructive Environment**

It is important to create a constructive environment so that right people are involved in the decision making process. Also this makes sure that team members and stake holders don't hesitate to participate.

- **Generate Good Alternatives**

Once you create constructive environment, conduct sessions of brainstorming. At this stage it is important not to discard any point of view and give equal importance to each idea. You can also have written ideas in short surveys from the team. This gives chance to quieter people to voice their opinion. Once ideas are generated – organize ideas with common theme in affinity groups category.

- **Look at High-Risk Consequences**

In some cases the impact of the decision may be significant. Analyze the consequences of each decision and look at its implications. In some cases it might be high on budget; some decision might require more people. Compare the decisions based on the available resources and constraints.

- **Consider Interpersonal Issues**

It can be difficult to predict how other people will react to the decision you make but at the same time it is important to understand the people acceptance on the implication of your decision

- **Choose the Best Alternative**

There are various tools available to help you with choosing the best alternative out of the available options. Please see the tools sections to refer to them.

- **Check Your Decision**

Make sure you save a session for checking all the assumptions and methods you used to arrive at the decision

- **Communicate Your Decision, and Take Action**

Once you have arrived at your decision, make sure you communicate it effectively to all the stake holders and involved parties. Let them know why you choose this decision and what expected impact and involved risks are. And importantly what projected benefits are. People involvement is the key to implementation. And last but most important thing is “act on your decision”.

Various Techniques for Decision Making

Few of the available methodologies are outlined below:

1. Grid Analysis Methodology

Grid Analysis is a particularly powerful tool for decision making where you have a number of good alternatives to choose from and when you have many different factors to take into account. Grid Analysis is the simplest form of Multiple Criteria Decision Analysis (MCDA)

How to Use the Tool As the name conveys, we need to make a Grid. Grid is comprised of options we have available and factors which affect our decision making.

Step 1

On the table list all of your ‘options’ as the row labels, and list the ‘governing factors’ as the column headings. For example, if you were going for a vacation, factors to consider might be cost of flight, cost of hotel, distance, activities, Quality and safety.

Step 2

Work your way down the columns of your table, scoring each option for each of the factors in your decision. Score each option from 0 (poor) to 5 (very good). Note that each score is independent of each other, so you do not have to have a different score for each option – if none of them are good for a particular factor in your decision, then all options should score 0

Step 3

The next step is to work out the relative importance of the factors in your decision. Show these as numbers from, say, 0 to 5, where 0 means that the factor is absolutely unimportant in the final decision, and 5 means that it is very important. Again, since many factors can have same importance, it’s not required to score it differently

Step 4

To get weighted scores for each options multiply each of your scores from step 2 by the values for relative importance of the factor that you calculated in step 3.

Step 5

Add up these weighted scores for each of your options. The option that scores the highest is the best suited option. Example: Say you have a situation where you are trying to decide the best vacation destination from the various options available to you.

In this case say the factors that you want to consider are:

Cost of travel, quality, distance, cost of hotel, activities and safety

Destination options are Hawaii, Disney, Vegas, and Maine

Grid 1 with options and factors and score for each factor

Factors	Travel Cost	Quality	Distance	Cost of Stay	Activities	Safety	Total
Weights	4	2	1	4	4	2	
Hawaii	1	3	1	3	4	3	
Disney	1	3	1	3	4	3	
Vegas	3	3	2	3	2	2	
Maine	4	4	2	2	0	4	

Next is to decide the relative weight for each factor and multiply these by the scores already entered and total them each.

Grid 2:

Factors	Travel Cost	Quality	Distance	Cost of Stay	Activities	Safety	Total
Weights	4	2	1	4	4	2	
Hawaii	4	6	1	12	16	6	45
Disney	4	8	2	8	20	8	50
Vegas	12	6	2	12	8	4	44
Maine	16	8	2	8	0	8	42

As evident by the grid results, Disney comes out as the winner!!

2. T-Chart Technique A T-Chart in the simplest form can be understood as a listing of positives and negatives surrounding a particular choice. Drawing up such a chart insures that both the positive and negative aspects of each direction or decision will be taken into account. For example, what are the pros and cons of deciding to hire more employees?

PRO	CON
Faster Code Development	Higher Expenses
Less Reliability on 1 person	Less Accountability
Expertise of Ideas	People Management Issues

3. PMI Technique T Chart method could be extended to include the aspects of decision which are outside the present context of judgment. Edward de Bono named it as PMI method for plus, minus, and interesting. In this method, you list all the good points of the idea, then all the bad points, and finally all the interesting points which include areas of curiosity or uncertainty, or attributes that you simply don't care to view as either good or bad at this point. For example, consequences that some people/or some situations might view as good and others might view as bad

Problem Solving

Problems are only opportunities in work clothes. – Henry Kaiser

EDIT NOTE: Add graphic re: problem solving

As a BA you are often solving a problem for a client (internal or external), supporting those who are solving problems, or discovering new problems to solve. Therefore, it's useful to get used to an organized approach to problem solving and decision making. Here are some guidelines to get you started. After you've practiced them a few times, they'll become second nature to you -- enough that you can deepen and enrich them to suit your own needs and nature.

Steps for Effective Problem Solving:

Clear Problem Definition: It's very important to have a precise, clear problem definition before starting any other work. Understand the issue first and then jump into solution mode. Challenge the definition from all angles: The more ways you can define a problem, the more likely it is that you will find the best solution. For example, "sales are too low" may mean strong competitors, ineffective advertising, or a poor sales process. Identify root cause: This is all about finding the root cause, rather than treating a symptom. If you don't get to the root, the problem will likely recur, perhaps with different symptoms. Don't waste time re-solving the same problem.

Approach with Multiple Solutions: The more possible solutions you develop, the more likely you will come up with the right one. The quality of the solution seems to be in direct proportion to the quantity of solutions considered in problem solving. Prioritize: Of all the available solutions, some might be more complex demanding more money or more time. You need to prioritize your solution based on its balance on short and long term impact.

Make a Decision: Once problem has been identified and solutions, you need to select one solution and act on it. The more you put off on deciding your best solution, the higher the cost and larger the impact of problem becomes. Best practice is to set a deadline for making a decision and course of action.

Identify Responsible People and Delegate: It's important to identify resources to carry on the task and make them responsible for different elements. Problem solving always needs multiple heads and hands working together.

Define Objective Measures of Success: In a long term and complex problem solving it is easy to get sidetracked. Set defined measurements and test for measurements of completion.

Techniques /Tools for Problem Solving:

5 Whys Tool Made popular in the 1970s by the Toyota Production System, the 5 Whys is a simple problem-solving technique that helps you to get to the root of a problem quickly.

How to Use the Tool:

When you're looking to solve a problem, start at the end result and work backward (toward the root cause), continually asking: "Why?" You'll need to repeat this over and over until the root cause of the problem becomes apparent.

Example: In this example, the problem is that your Business Unit is unhappy. Using the 5 Whys, you go through the following steps to get to the cause of the problem:

1. Why is our Business Unit unhappy? Because developers didn't deliver the services, when they said they would.
2. Why were developers unable to meet the agreed-upon timeline or schedule for delivery? The job took much longer than we thought it would.
3. Why did it take so much longer? Because developers underestimated the complexity of the job.
4. Why did we underestimate the complexity of the job? Because developers are not so experienced and are running short on staff

Solution/Conclusion: need to have more experienced developers / Need to hire more developers / divide work load evenly and have realistic time estimates

Benefits of the 5 Whys include:

- It helps you to quickly determine the root cause of a problem.
- It's simple, and easy to learn and apply.

2. Constructive Controversy:

Involving other people – who inevitably have different perspectives and views – helps us ensure that we've considered solutions from all possible sides. This problem-solving approach was introduced by David Johnson and Roger Johnson in 1979. This has been recognized as a leading model for problem solving by Arguing For and Against Your Options

How to do it:

- You need to facilitate meetings where people come and give constructive criticism to identified solutions
- For all rounded approach it's important to understand that people agree to disagree gracefully. It's not about "winning" but rather it's an approach to improve collective understanding.
- Listen actively, and ask for clarification when necessary
- Commit to understanding all sides of an issue.

Creativity

Creativity and Its Role in Business Analysis

Creativity Techniques

Procurement, Including RFPs, RFIs and IFBs

Types of Procurements (RFx)

One role typically facilitated by business analysts within an organization is that of defining business and system requirements used in procurement.

There are a number of different types of procurements, and they each have unique features. Procurement guidelines vary from state to state (and country to country), so it is generally best to consult your finance and legal offices for specific dos and don'ts. The material covered here is generic in nature to ensure applicability across all organizations.

The type of purchase methodology generally depends on the types of goods and services being procured. Below are the most common forms:

Type of Procurement	Description
Invitation for Bid (IFB)	Typically the goods or services being tendered are commercially available from more than one vendor. In this type of procurement, specifications are provided and the responsible bidder with the best price is awarded a contract. A company must be able to clearly define specifics, including terms and conditions, in the solicitation.
Request for Information (RFI)	When companies or governments are considering procurement, they may start with a request for information from various vendors in order to better frame their procurement. The RFI is typically not binding, and result in specific information, versus a quotation. Generally when done, this step occurs before a formal IFB/RFQ or RFP.
Request for Quotation (RFQ)	This is the same as an IFB.
Request for Proposal (RFP)	When a custom solution is desired, often involving hardware, software and/or system development and ongoing maintenance, a request for proposal is issued. This results in a proposal outlining how the bidder firm will meet the requirements. Generally these take the longest to complete, but are very helpful when introducing new technologies or systems that a firm wishes to outsource.

Some entities opt to try demonstration projects or proofs of concept on some technologies before making significant investments. These types of "try before you buy" techniques can also be incorporated into the procurement.

All but the RFI typically results in some sort of procurement to the winning bidder(s), whether it be a purchased order (PO), in the case of a IFB/RFQ, or a contract, in the case of an RFP. Depending upon the request, the award of a PO or contract can be to one or more vendors.

Make vs. Buy

Business and governmental agencies face the decision whether to “make or buy” on a regular basis, and typically the analysis is conducted by a business analyst. Many factors go into these decisions, namely:

- Is this part of our current (or desired) core competence
- Do we have the necessary skills, time and capacity to build in house
- Can we develop more cost effectively than buying
- Number and reliability of suppliers in the marketplace
- Ongoing maintenance costs if we build versus likely inflation costs of suppliers in the future
- Ability to easily scale (up or down) as demand occurs in the future
- Ability to reuse or recycle components/parts into other products and services
- Authority to contract out the products/services
- Opportunity costs of the decision (e.g. if resources diverted to build, what do we no longer have the capacity to do)

While not an exhaustive list—the one above does cover some of the more important ones to consider. In recent years, the decision whether to make or buy has turned gray! There is often a middle of the road option, called a hybrid, which has a little bit of both as a third alternative. Below is a hypothetical example which outlines the three alternatives:

Make	Hybrid	Buy
Large custom software build using internal resources	Purchase of configurable software; tweaking by internal resources	Outsource development and ongoing maintenance to a 3rd party
Integrate functionality into current systems	Use of external experts in the software to facilitate knowledge transfer for maintenance	Use of 3rd parties to assist with change management (staff training & materials)
Approx 24-36 months to do	Approx 12-24 months to do	Approx 9-12 months to do

Hybrid solutions can be partnering and collaboration with other parties—whether it be purchase cooperative, private-public partnership, or any variation that leverages external parties to achieve the organizational mission.

Procurement Guidelines

While these will vary by the entity letting the procurement, elements of contract law, procuring the best value for the organization, and ability to withstand scrutiny are common across all jurisdictions. Generally, these include items such as:

- Posting the offering so that the opportunity is known to potential bidders (in NYS this entails posting in the NYS Contract Reporter)
- Shielding those doing the procurement from influence by potential bidders (in NYS an active procurement goes into a quiet period, bidders must avoid discussing the current offering except where allowed by law, and all contact must be noted and reported where required)
- Allowing sufficient time for bidders to put together a quote or a proposal (in NYS, these timeframes are detailed in procurement guidelines)
- Security on the part of the bidders in the form of bid bonds, letters of credit, and assurances from the corporate executives that the company will meet requirements if awarded the bid
- Development of the evaluation methodology in advance of the bids. In NYS the categories, where applicable, are published with the values assigned, e.g. cost = 25%, technical proposal = 50%, references = 10%, etc.)
- Acknowledgement that submission of the bid creates a binding price/offer for a specified duration.

The terms and conditions specific to the procurement are generally detailed as part of the RFP/RFQ issued.

Best Practices and Timelines

The role of the business analyst is critical to a successful outcome. It is generally not possible to add additional items during negotiations and contracting, so carefully detailing the specifications into a statement of work is the key to a successful contract down the road. Below are some techniques that have helped to facilitate successful procurements:

- Clearly understand the industry from which you are soliciting from. Are there industry standards to consider,

Business Intelligence and Performance Management

Performance Management

Thank you Jim!

Requirements Templates

Business Requirements Documents

links to pb.works.com templates

Glossary

Business Analysis Glossary

A

Activity diagram A type of flowchart, part of the UML standard, that depicts activities, their sequence, and the flow of control.

Analysis paralysis An informal phrase applied to when the opportunity cost of decision analysis exceeds the benefits. In software development, analysis paralysis manifests itself through exceedingly long phases of project planning, requirements gathering, program design and modeling, with little or no extra value created by those steps.

Artifact

As-is modeling Refers to gathering information about the current state of the business area being analyzed; e.g., current processes and data.

Assumptions and Constraints Assumptions and constraints identify aspects of the problem domain that are not functional requirements of a solution, and will limit or impact the design of the solution.

B

BA Business Analyst or Business Analysis

BA Bok Business Analysis Body of Knowledge

BA-COP Business Analysis Community of Practice

BA CoE Business Analysis Center of Excellence

BP/LL Best Practices/Lessons Learned

BRD Business Analysis Requirements Document

Business analysis Business analysis is the set of tasks, knowledge, and techniques required to identify business needs and determine solutions to business problems. Solutions often include a systems development component, but may also consist of process improvement or organizational change.

Business analyst A business analyst works as a liaison among stakeholders in order to elicit, analyze, communicate and validate requirements for changes to business processes, policies and information systems. The business analyst understands business problems and opportunities in the context of the requirements and recommends solutions that enable the organization to achieve its goals.

Business area A naturally cohesive group of activities that share data and may cross organizational boundaries.

Business case The first document completed during project initiation; it contains the analysis and results of business assessments providing the justification to pursue a project opportunity.

Business process improvement Analyzing a business process to increase its efficiency and effectiveness; identifying and eliminating causes of poor quality, process variation, and non-value-added activities.

Business requirements Higher-level statements of the goals, objectives, or needs of the enterprise. They describe such things as the reason a project was initiated and the things the project will achieve.

Business requirements document The document or artifact that captures gathered requirements and serves to communicate them.

C

Candidate solution A suggestion that might be a good thing to do.

Context diagram Represents the entire system as if it were a single process; often used to determine a system boundary or solution scope with stakeholders or SME's.

Customer The business units at all levels of the organization that identified the need for the product or service the project will develop.

D

Data flow diagram Graphical representation of the flow of data through a manual or automated information system which illustrates the processes, data stores, and external entities in a business or other system and the connecting data flows.

Data model A model showing data requirements of a business area. An Entity Relationship Diagram is a common type of data model.

E

Elicit The process of drawing requirements from stakeholders and SME's.

F

Features A service a solution provides to meet a need. Typically high-level abstractions of solution that will turn into functional or non-functional requirements. They allow for early prioritization and scope management, and for getting a high-level sense of the stakeholders view of the solution.

Functional requirements

What the systems/products are, do, or provide from the customer's point of view.

- G -

- H -

- I -

J

JAD Joint application development

- K -

- L -

- M -

N

Non-functional requirements Requirements that do not directly relate to the behavior or functionality of the solution, but rather describe environmental conditions under which the solution must remain effective or qualities that the systems must have.

O

Object-oriented modeling An approach to business or software engineering in which components are made up of encapsulated groups of data and functions which can inherit behavior and attributes from other components, and whose components communicate via messages to one another.

P

Pains and pleasures Pain – something that is currently causing frustration, lack of productivity, poor quality, etc.
Pleasure – something that is currently working well. It would be good to continue it.

Product scope The functions or features that characterize a product or service.

Project scope The work that must be done to deliver a product with the specified features and functions.

Q

QA Quality assurance

QC Quality control

R

Requirements Conditions or capabilities needed by a stakeholder to solve a problem or achieve an objective.

RFP Request for proposal

RFQ Request for quotations

Requirements work plan Documents the requirements activities a business analyst will perform on a particular project, what deliverables they will produce, and how they will control and manage changes to the deliverables.

S

Schedule Planned dates for performing activities and for meeting deliverables.

Scope Creep Gradual addition of new requirements to the original product specifications.

SDLC System development life cycle

Subject matter expert (SME) A person expert in the particular business area being analyzed. SME's are often the primary source of requirements for a business analyst.

Solution Something that fills a business need. Solutions do not always involve automation; they could involve streamlining a business process, etc.

Statement of work A narrative description of products or services to be supplied under contract.

Stakeholder Individuals and organizations that are involved in, or may be affected by, project activities.

T

To-be modeling Refers to analyzing and documenting the potential future state of the business area being analyzed; processes, data, activities, how the activities would be accomplished.

Traceability The ability to map solution components and requirements backward and forward through the development cycle.

U

Unified modeling language (UML) An object-oriented modeling language governed by the Object Management Group (www.omg.org).

Use case diagram An analytical tool consisting of text and models that describes the tasks a system will perform for actors, and the goals the system achieves for those actors along the way.

User acceptance testing User Acceptance Testing is typically the final phase in a software development process in which the software is given to the intended audience to be tested for functionality. UAT is also called beta testing, end-user testing or application testing.

V

Validate Reviewing requirements to ensure they are correct.

W

WBS Work breakdown structure

Workflow model Describes the tasks, decisions, inputs and outputs, people and tools involved in a specific process.

- X -

- Y -

- Z -

Resources

Noted Contributors

The material included in this wikibook was authored by members of the Capital District Business Analysis Community of Practice (BACOP) Guidebook Committee, led by Barbara Ash (retired from the NYS Office of the State Comptroller) and Kelly Smith-Lawless (currently PPM Director of the General Gov't Cluster within NYS Office for Information Technology). This material was organized and drafted over a period of a couple years and care was taken to ensure representation from across various agencies within NYS Government. The goal was to establish a common framework from which all IT projects could leverage in order to develop a consistent approach across different agencies. BACOP Representatives (past and present) that contributed to this wikibook as authors and editors include:

- Jim Alderdice (ITS)
 - Anonymous (HRI)
 - Barbara Ash (OSC)
 - Marina Brenton (Health)
 - Todd Britton (OSC)
 - Karen Connors (PubSafety)
 - Michelle Cuchelo (GGC)
 - Susan Davis (NYSIF)
 - Marcia Eaton (DOH)
 - Belinda Jackson (Health)
 - Deena Jones (WCB)
 - Erlene Kent (TED)
 - Tina Koberger (TED)
 - Nathan Kroop (Health)
 - Kimberly Manion (Health)
 - Prachi Mondaiyka (Health)
 - Beth Ostwald (Tax)
 - Kelly Smith-Lawless (GGC)
 - Susan Travis (GGC)
 - Elaine Wing (Health)
-

- Elaine Zuk (ENV)

Contributors

List of Figures

Licenses

Two copyright licenses are included here in their entirety to permit usage by users of this wikibook. Links are included for the most current versions, in the event they are updated.

1. Creative Commons Attribution-Share Alike 3.0 Unported License
2. GNU Free Documentation License

Creative Commons Attribution Unported License

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

- a. "**Adaptation**" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.
 - b. "**Collection**" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each
-

constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined below) for the purposes of this License.

- c. "**Creative Commons Compatible License**" means a license that is listed at <http://creativecommons.org/compatiblelicenses> that has been approved by Creative Commons as being essentially equivalent to this License, including, at a minimum, because that license: (i) contains terms that have the same purpose, meaning and effect as the License Elements of this License; and, (ii) explicitly permits the relicensing of adaptations of works made available under that license under this License or a Creative Commons jurisdiction license with the same License Elements as this License.
- d. "**Distribute**" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.
- e. "**License Elements**" means the following high-level license attributes as selected by Licensor and indicated in the title of this License: Attribution, ShareAlike.
- f. "**Licensor**" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.
- g. "**Original Author**" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.
- h. "**Work**" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.
- i. "**You**" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.
- j. "**Publicly Perform**" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.
- k. "**Reproduce**" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights

Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant

Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

- a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;
- b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified.";
- c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,
- d. to Distribute and Publicly Perform Adaptations.
- e. For the avoidance of doubt:
 - i. **Non-waivable Compulsory License Schemes.** In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;
 - ii. **Waivable Compulsory License Schemes.** In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,
 - iii. **Voluntary License Schemes.** The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions

The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

- a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(c), as requested. If You create an Adaptation, upon notice from any Licensor You

- must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(c), as requested.
- b. You may Distribute or Publicly Perform an Adaptation only under the terms of: (i) this License; (ii) a later version of this License with the same License Elements as this License; (iii) a Creative Commons jurisdiction license (either this or a later license version) that contains the same License Elements as this License (e.g., Attribution-ShareAlike 3.0 US); (iv) a Creative Commons Compatible License. If you license the Adaptation under one of the licenses mentioned in (iv), you must comply with the terms of that license. If you license the Adaptation under the terms of any of the licenses mentioned in (i), (ii) or (iii) (the "Applicable License"), you must comply with the terms of the Applicable License generally and the following provisions: (I) You must include a copy of, or the URI for, the Applicable License with every copy of each Adaptation You Distribute or Publicly Perform; (II) You may not offer or impose any terms on the Adaptation that restrict the terms of the Applicable License or the ability of the recipient of the Adaptation to exercise the rights granted to that recipient under the terms of the Applicable License; (III) You must keep intact all notices that refer to the Applicable License and to the disclaimer of warranties with every copy of the Work as included in the Adaptation You Distribute or Publicly Perform; (IV) when You Distribute or Publicly Perform the Adaptation, You may not impose any effective technological measures on the Adaptation that restrict the ability of a recipient of the Adaptation from You to exercise the rights granted to that recipient under the terms of the Applicable License. This Section 4(b) applies to the Adaptation as incorporated in a Collection, but this does not require the Collection apart from the Adaptation itself to be made subject to the terms of the Applicable License.
- c. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv) , consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(c) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.
- d. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.
-

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability

EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

- a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.
- b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

- a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
 - b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
 - c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
 - d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
 - e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
 - f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms
-

Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of the License.

Creative Commons may be contacted at <http://creativecommons.org/>.

Version 1.3, 3 November 2008 Copyright (C) 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc.
<<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

GNU Free Documentation License

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
 - B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
 - C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
 - D. Preserve all the copyright notices of the Document.
 - E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
 - F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
-

- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified version.
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME.

Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.3
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
A copy of the license is included in the section entitled "GNU
Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the
Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Article Sources and Contributors

Business Analysis Guidebook *Source:* <https://en.wikibooks.org/w/index.php?oldid=2597198> *Contributors:* KellyLawless, 2 anonymous edits

Introduction *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608502> *Contributors:* KellyLawless

The Business Analyst Role *Source:* <https://en.wikibooks.org/w/index.php?oldid=2595662> *Contributors:* KellyLawless

Keys and Barriers to Business Analyst Success *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608503> *Contributors:* KellyLawless

Maturity Models for Business Analysis and Self-Assessment *Source:* <https://en.wikibooks.org/w/index.php?oldid=2597420> *Contributors:* KellyLawless

Stakeholder Analysis *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608505> *Contributors:* KellyLawless

Building a Business Case *Source:* <https://en.wikibooks.org/w/index.php?oldid=2595756> *Contributors:* KellyLawless

Documenting and Managing Requirements *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608204> *Contributors:* CommonsDelinker, Ethacke1, KellyLawless

Requirement Gathering Tools *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608115> *Contributors:* Ethacke1, KellyLawless

Business Analysis Within a Typical System Development Life Cycle *Source:* <https://en.wikibooks.org/w/index.php?oldid=2595700> *Contributors:* KellyLawless

Data Modeling and Data Documentation *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608281> *Contributors:* Ethacke1

System Acceptance, Test Planning and Strategy *Source:* <https://en.wikibooks.org/w/index.php?oldid=2595703> *Contributors:* KellyLawless

Implementation and Training *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608507> *Contributors:* KellyLawless

User Experience *Source:* <https://en.wikibooks.org/w/index.php?oldid=2607109> *Contributors:* Kmpalmo, 11 anonymous edits

Business Process Improvement Models *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608508> *Contributors:* KellyLawless

Root Cause Analysis *Source:* <https://en.wikibooks.org/w/index.php?oldid=2597405> *Contributors:* KellyLawless

Strategic Analysis and Planning *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608510> *Contributors:* KellyLawless

LEAN *Source:* <https://en.wikibooks.org/w/index.php?oldid=2597454> *Contributors:* KellyLawless

Facilitation and Elicitation Techniques *Source:* <https://en.wikibooks.org/w/index.php?oldid=2597446> *Contributors:* KellyLawless

Prioritization Techniques *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608511> *Contributors:* KellyLawless

BA Software Tools *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608514> *Contributors:* KellyLawless

Communication Skills *Source:* <https://en.wikibooks.org/w/index.php?oldid=2595765> *Contributors:* KellyLawless

Building Relationships and Trust *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608515> *Contributors:* KellyLawless

Analytical Thinking and Problem Solving *Source:* <https://en.wikibooks.org/w/index.php?oldid=2597408> *Contributors:* KellyLawless

Creativity *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608518> *Contributors:* KellyLawless

Procurement, Including RFPs, RFIs and IFBs *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608519> *Contributors:* KellyLawless

Business Intelligence and Performance Management *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608521> *Contributors:* KellyLawless

Requirements Templates *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608522> *Contributors:* KellyLawless

Glossary *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608523> *Contributors:* KellyLawless

Resources *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608524> *Contributors:* KellyLawless

Noted Contributors *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608237> *Contributors:* KellyLawless

Contributors *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608525> *Contributors:* KellyLawless

List of Figures *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608526> *Contributors:* KellyLawless

Licenses *Source:* <https://en.wikibooks.org/w/index.php?oldid=2608230> *Contributors:* KellyLawless

Image Sources, Licenses and Contributors

File:Requirement Attributes.png *Source:* https://en.wikibooks.org/w/index.php?title=File:Requirement_Attributes.png *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:Ethacke1

File:ChangeRequestAttributes.jpg *Source:* <https://en.wikibooks.org/w/index.php?title=File:ChangeRequestAttributes.jpg> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:ChangeManagementProcess.png *Source:* <https://en.wikibooks.org/w/index.php?title=File:ChangeManagementProcess.png> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:Ethacke1

File:Forward Traceability.PNG *Source:* https://en.wikibooks.org/w/index.php?title=File:Forward_Traceability.PNG *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:Ethacke1

File:RequirementTraceabilityMatrix.jpg *Source:* <https://en.wikibooks.org/w/index.php?title=File:RequirementTraceabilityMatrix.jpg> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:HighLevelProcessToPurchaseACar.jpg *Source:* <https://en.wikibooks.org/w/index.php?title=File:HighLevelProcessToPurchaseACar.jpg> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:ExpandedCreateOfferSubProcess.jpg *Source:* <https://en.wikibooks.org/w/index.php?title=File:ExpandedCreateOfferSubProcess.jpg> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:TransferInsuranceUseCase.jpg *Source:* <https://en.wikibooks.org/w/index.php?title=File:TransferInsuranceUseCase.jpg> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:PurchaseOfferStoryboard.jpg *Source:* <https://en.wikibooks.org/w/index.php?title=File:PurchaseOfferStoryboard.jpg> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:ScreenShotMockUp.jpg *Source:* <https://en.wikibooks.org/w/index.php?title=File:ScreenShotMockUp.jpg> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:OwnerInfoWireframeScreen.jpg *Source:* <https://en.wikibooks.org/w/index.php?title=File:OwnerInfoWireframeScreen.jpg> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:OwnerVehicleDataDictionary.jpg *Source:* <https://en.wikibooks.org/w/index.php?title=File:OwnerVehicleDataDictionary.jpg> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:OwnerForm.jpg *Source:* <https://en.wikibooks.org/w/index.php?title=File:OwnerForm.jpg> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:VisioDropandDragShape.jpg *Source:* <https://en.wikibooks.org/w/index.php?title=File:VisioDropandDragShape.jpg> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:EditingVisioShapes.jpg *Source:* <https://en.wikibooks.org/w/index.php?title=File:EditingVisioShapes.jpg> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:ConnectingVisioShapes.jpg *Source:* <https://en.wikibooks.org/w/index.php?title=File:ConnectingVisioShapes.jpg> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:FishboneDiagram.jpg *Source:* <https://en.wikibooks.org/w/index.php?title=File:FishboneDiagram.jpg> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:Fishbone BadCoffeeExample.jpg *Source:* https://en.wikibooks.org/w/index.php?title=File:Fishbone_BadCoffeeExample.jpg *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:Root Cause Analysis Tree Diagram.jpg *Source:* https://en.wikibooks.org/w/index.php?title=File:Root_Cause_Analysis_Tree_Diagram.jpg *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:RootCauseParetoChart.jpg *Source:* <https://en.wikibooks.org/w/index.php?title=File:RootCauseParetoChart.jpg> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:KellyLawless

File:ValueStreamMapParts.png *Source:* <https://en.wikibooks.org/w/index.php?title=File:ValueStreamMapParts.png> *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* User:DanielPenfield

File:Tips for Facilitation.png *Source:* https://en.wikibooks.org/w/index.php?title=File:Tips_for_Facilitation.png *License:* Creative Commons Attribution-Sharealike 3.0 *Contributors:* anne kehlhofer

License

Creative Commons Attribution-Share Alike 3.0
[//creativecommons.org/licenses/by-sa/3.0/](https://creativecommons.org/licenses/by-sa/3.0/)
