4.0

Stakeholders Needs Elicitation

Reference Guide

enFOCUS SOLUTIONS

V. 1

Table of Contents

Table of Contents Cont'd

Introduction

"When people talk, listen completely. Most people never listen."—Ernest Hemingway

It's incredibly difficult to gather a complete and well-written set of solution requirements from business and technical stakeholders. Instead of attempting to get these requirements direct from stakeholders, best practice is for the business analyst (BA) to elicit a set of information that can be used to develop functional and non-functional requirements. This information is commonly referred to as user requirements, or stakeholder needs.

Gathering stakeholder needs is a crucial task which will provide the BA with sufficient information to develop solution requirements. It's difficult to get well-written requirements directly from stakeholders, so many BAs have adopted the practice of eliciting information that tells the BA what the stakeholders need, and then turning that information into concrete requirements that can be delivered to developers to be built.

But before the BA gathers stakeholder needs, s/he must prepare for elicitation by determining which techniques to use, as well as the schedule and subsequent details. Preparing for elicitation before conducting events will help to minimize the amount of problems encountered during requirements development, management, and implementation.

In addition to direct input from stakeholders, the BA will rely on other sources of information to develop a complete set of requirements. This information includes areas such as rule books, terminology glossaries, and assumptions. All of these elements, documented in RequirementPro™, will have an important role in the next task group, *5.0 Requirements Development.*

4.1 Chapter One
Prepare for Elicitation

Introduction

At this point in your project, two very important milestones have been reached as a result of performing the tasks in the previous task group, *3.0 Solution Conceptualization:*

- The project team agreed upon a set of features that will guide the development, construction, and implementation of a successful final solution.
- Project approval was gained with the business case and the project team now has the go-ahead to begin solution development.

With those milestones achieved, the team is ready to begin eliciting information from stakeholders that will lead to the development of formal requirement specifications. Much of the elicitation process is spent preparing for elicitation events. Before attempting to gather stakeholder needs, focus on planning the rest of *Task Group 4.0 Stakeholder Needs Elicitation:*

- 4.2 Gather Stakeholder Needs: Determine the best process for eliciting needs.
- 4.3 Gather and Analyze Documents: Determine the types of documents that will be useful.
- 4.4 Gather Business Rules: Determine which rule books need to be created or revised.
- 4.5 Document Terminology: Determine what types of terms you'll need to research and document.
- 4.6 Gather Assumptions: Determine the process for gathering and documenting assumptions.

Core Concepts

Elicitation

Elicitation is the process of identifying sources for requirements and then extracting requirements from those sources. This task involves the collection of information from stakeholders by developing human relationships. Remember that requirements elicitation is a "human-intensive" activity that relies on stakeholder knowledge as a primary source of information. Eliciting needs *directly* from stakeholders is a crucial part of elicitation. Failure to do so often results in less than optimal results or a failed product, as lack of user input is the number one cause for failed or challenged projects.

At this stage in the requirements development process, the business analyst must elicit user requirements, or *stakeholder needs*, to serve as the basis for the more formal solution requirements development that occurs in the next task group of the Requirements Excellence Framework™. A *stakeholder need* is a type of requirement known as a user requirement, which consists of a medium level of detail that specifies what the user needs in terms of the system. Stakeholder needs exist in Enfocus Requirements Suite™ in the form of Scenarios and Stakeholder Needs Patterns (available in the Portfolio Performance Subscription). For more information on these two concepts, refer to the next chapter, *Task 4.2 Gather Stakeholder Needs.*

Minimize the Amount of Problems

Organizations are looking to minimize the amount of problems that commonly appear in requirements elicitation, such as stakeholders that aren't participating or getting along with others. Adequate preparation for stakeholder needs elicitation will prevent problems before they even occur. Prior to starting elicitation, the BA usually reviews the information documented in previous elicitation sessions, as well as the information gathered in stakeholder analysis. Understanding how elicitation has gone before, as well as understanding the stakeholders you're working with, will help to minimize the risk of problems arising. Beware of the following factors that may prove to be a source of future problems among stakeholders:

- Inadequate communication
- Time differences or separations
- Cultural differences
- Knowledge management inadequacies

It is a good practice to analyze the possible sources of problems in each iteration of the elicitation process, because problems can vary throughout the project lifecycle. Depending on the features in the iteration you're eliciting for, stakeholders will most likely change.

Process

Most of the preparation for elicitation is carried out by the business analyst (BA). However, s/he will most likely require some input from relevant stakeholders with subject matter knowledge, to ensure elicitation gets planned properly.

Before performing the process of gathering needs, you need to identify the specific individuals and documents from which you will elicit the information required to develop solution requirements. This activity will be easy if the BA correctly performs *Task 3.2 Conduct Stakeholder Analysis*. With the Portfolio Performance Subscription of Enfocus Requirements Suite™, an easy and quick way to determine which stakeholders should be involved is to review the impacted *stakeholder personas* in RequirementPro™. Although stakeholder analysis is not automated with the basic subscription of Enfocus Requirements Suite™, it is still a vital task that will produce useful information for the entire project lifecycle.

Then, ensure the following ten steps are performed for elicitation preparation. While it makes sense to perform some steps subsequent to others, it doesn't really matter in what order you perform these steps, as long as all tasks are completed.

1. Determine the frequency of elicitation (All in one go? Or iteratively?).
2. Determine the stakeholders to be involved.
3. Determine the best elicitation techniques.
4. Determine the elicitation deliverable.
5. Determine the schedule for elicitation.
6. Ensure stakeholders understand business requirements.
7. Determine document types to be gathered.
8. Determine rule books to be gathered.
9. Determine terms to be documented.
10. Determine assumptions to be documented.

1. Determine the Frequency of Elicitation.

How often will elicitation occur in this project? The answer to this question will vary according to the project's development style. Waterfall projects usually require elicitation activities to be performed all at once, whereas Agile shops usually perform elicitation iteratively. To determine the frequency of elicitation, think about the decisions made in *Task 3.6 Define Solution Features.* If the team agreed to develop features iteratively, then you will also need to elicit needs iteratively; if the team agreed to define all features up front in the project, then you'll be eliciting needs all in one go.

Once you understand when elicitation will occur, you can begin to plan the specifics, like the people that will be involved, the techniques that you'll use, and the schedule of events.

2. Determine the Stakeholders to Be Involved.

Compare the information acquired in the previous step and in *Task 3.2 Conduct Stakeholder Analysis* to determine which of the stakeholders will be the best sources of information. Usually, individual stakeholders are related to specific features. Knowing which features will need requirements first will help you to determine which stakeholders you need to talk to first.

Start off considering all of the project stakeholders, including the project sponsor or customer, the users who will interact directly or indirectly with the software, compliance officers and internal audit, user support, help desk resources, and development resources such as enterprise architects, DBAs, developers, etc.

Then, divide them into categories according to their relationship with requirements: suppliers, receivers, and supporters. Organizing the set of stakeholders in this manner will help you decide which stakeholders you need to meet with or obtain information from. *Suppliers* are the ones that supply the business needs; *receivers* receive the requirements and must understand them in order to build the solution; and, *supporters* are the ones that generally pay for the solution. The table below helps to demonstrate this concept.

SUPPLIERS	RECEIVERS	SUPPORTERS
Stakeholders Project Sponsor Customers Users Business SMEs Business Process Experts Business Rules Experts Technical SMEs Internal Auditors Compliance Officers Market Analysts Legal Organizational Change Help Desk Documents Business Process Models Product Plans and Roadmaps Regulations Audit Reports	Development Team Members Designers System Architect QA and Testing Network Engineer DBA Data Warehouse Workflow Rules manager Business Continuity Security Configuration Management End User Training Business Rulebook Owners Maintenance and Support Staff Help Desk Technical Writers	Project Sponsor Business Process Owner CEO CFO CIO Project Investors

3. Determine the Best Elicitation Techniques.

As you prepare for elicitation, review the stakeholder analysis information and consider the best technique for each group. Some stakeholders may prefer an interview, while others may just want to enter their needs directly into StakeholderPortal™. Research and review the most effective elicitation techniques for your project, considering what your project team has learned about impacted stakeholders. This contextual information will dictate your approach. Keep in mind what you know about the stakeholders that are involved, and answer questions like the ones listed below to help determine the most suitable elicitation technique for the set of stakeholders that you're working with.

- Are stakeholders willing to learn a new technique?
- Are stakeholders familiar with StakeholderPortal™?
- Are stakeholders willing to learn how to use a new tool?
- Which elicitation techniques are commonly used in the organization?
- Which techniques are stakeholders most familiar with?

The analysts at Enfocus Solutions have found the techniques listed below to be useful in various types of projects. Refer to the descriptions that follow for an understanding of what each of these elicitation methods entails and how to prepare.

- Analysis of Existing Systems
- Direct Entry via StakeholderPortal™
- Elicitation Workshops
- Focus Groups
- Interviews
- Observation
- Surveys

Analysis of Existing Systems

Before interviews, observation sessions, etc., the stakeholder will expect that the analyst has already developed an understanding of the existing system. Analyzing existing related systems will help you take into account real usage patterns, human issues, common activities, and the relative importance of tasks/features. It will also point out obvious possible improvements, like features that are missing or do not currently work well. This technique is especially useful when the goal of the project is to build an improved version of an existing system. However, we suggest using this technique first before all others regardless of the type of project, unless no systems currently exist. Artifacts to review:

- System documentation
- Enhancement requests
- Problem logs

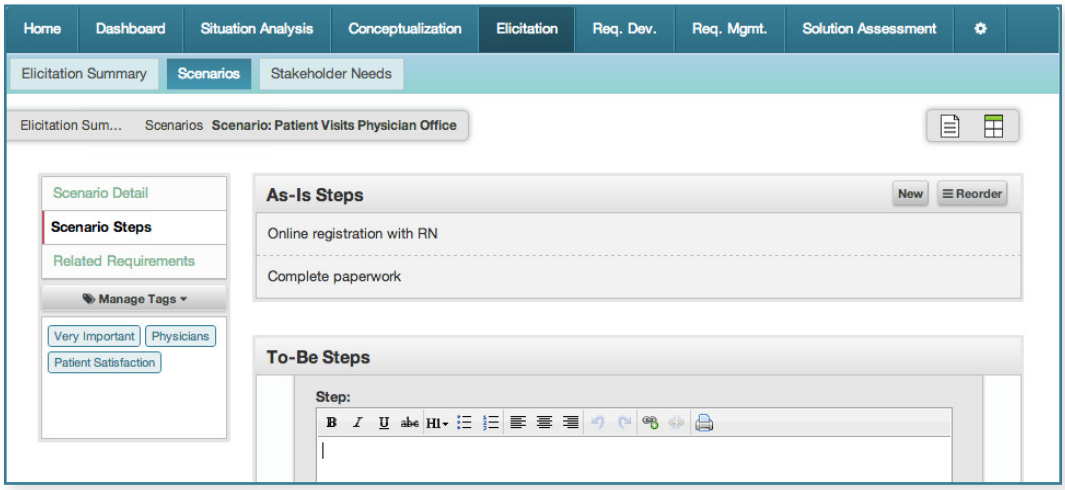
Before Using This Technique

- Determine how relevant the current system is to the future functionality.

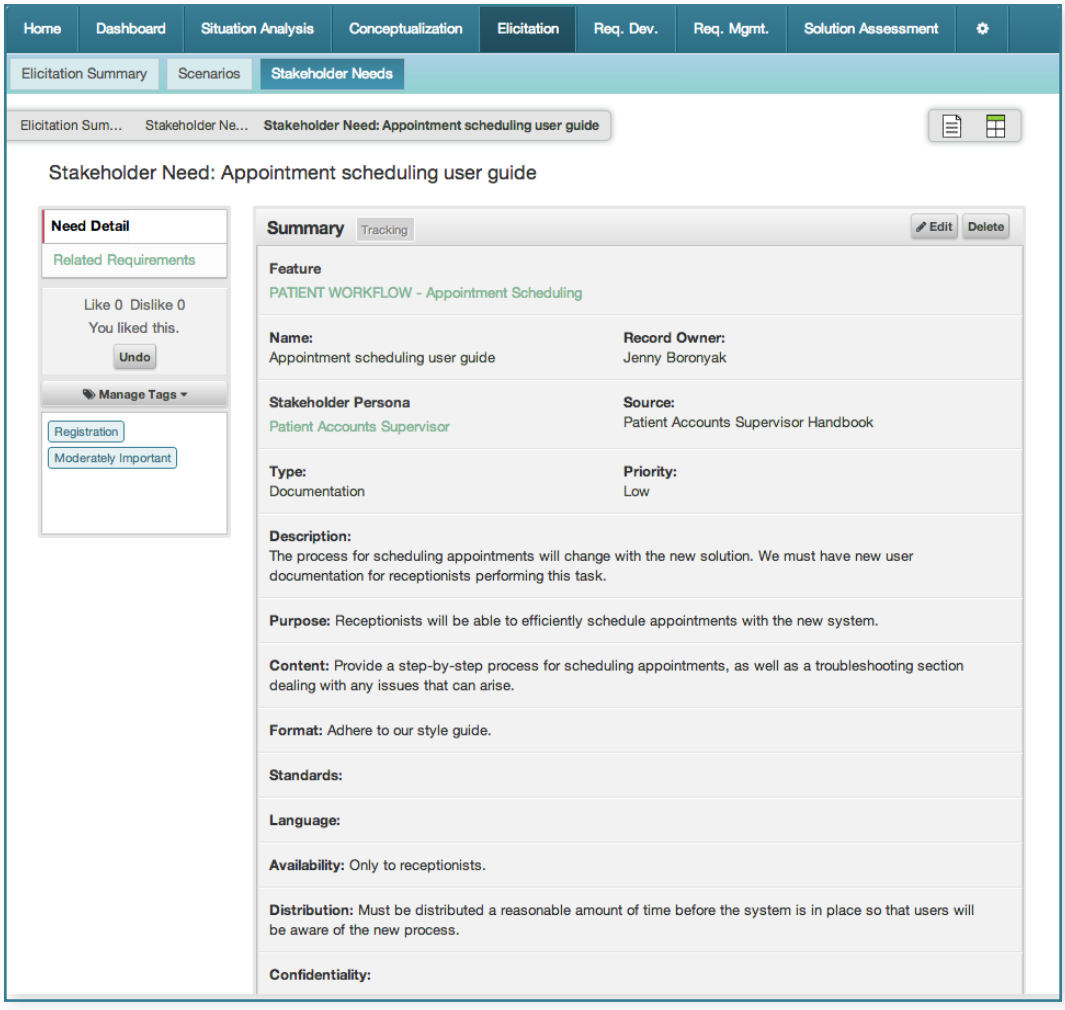
Direct Entry via StakeholderPortal™

One of the fundamental concepts behind RequirementPro™ and StakeholderPortal™ is the direct entry of needs by stakeholders. Using this intuitive approach to elicitation, a business analyst can save a lot of time and energy by having stakeholders input needs through the system via records such as Scenarios or Stakeholder Needs Patterns, if available with the Portfolio Performance Subscription. Another similar technique would be to use one of the other techniques mentioned in this chapter, followed by an analyst's input of data from an elicitation session into RequirementPro™. The former represents the quickest and most efficient method, but the latter is still incredibly effective.

Regardless of which technique you determine to be best, we suggest ensuring all needs have been entered into StakeholderPortal™ or RequirementPro™ before moving forward in the project lifecycle. The screenshots below display the views when a stakeholder enters a new need in StakeholderPortal™ via Scenarios and Stakeholder Needs Patterns (remember, patterns are only available in the Portfolio Performance Subscription). Don't forget that these records can also be entered on behalf of stakeholders in RequirementPro™. For more information on these features in RequirementPro™, see the next chapter.



Scenarios are developed in steps.



Patterns, available only with a Portfolio Subscription, consist of with pre-defined fields.

Before Using This Technique

- Ensure stakeholders have access to StakeholderPortal™ and related support channels.
- If necessary, facilitate StakeholderPortal™ training.

Elicitation Workshops

Facilitated *elicitation workshops* permit collaboration between analysts and stakeholders and are a powerful way to explore user needs and to draft requirements documents. In a requirements workshop, an experienced facilitator will guide the discussion by didactically presenting topics and delivering best practices to be covered in the meeting. This technique is particularly helpful in determining specific measures and initiatives for improvement processes. Workshops are most often used when there is a large group of stakeholders that must reach a consensus. It can take a significant amount of planning, training, and dedication of resources to pull off a successful workshop.

Before Using This Technique

Participants in a workshop can consist of everyone from stakeholders and users to product managers, subject matter experts, and testers. Make sure the right users participate. This means they have subject matter knowledge, experience in the organization, good reputation, and the ability to work in a group environment. Also, it's helpful to invite a cross-section of users. Although the experienced user will know what is important to the organization, the novice user is often better at explaining difficulties encountered in their work. There are several roles that need to be filled in an elicitation workshop:

- *Facilitator:* runs the meeting and ensures that it stays close to the agenda.
- *Scribe:* documents the results of the meeting.
- *Primary participants:* have speaking and voting rights.
- *Advisory participants:* have voting rights.
- *Observers:* do not have speaking or voting rights.

The *facilitator* is a key element to any workshop and will often have an effect on its success. Good facilitators can get an understanding of how well the team is functioning, guide them while they're struggling, and ensure participation by all team members present. The facilitator should be an unbiased individual with no interest in the project's results to ensure s/he is not viewed as partial to one stakeholder group over another. According to *Determining Project Requirements* by Hans Jonasson, the facilitator will play the following key roles:

- Timekeeper
- Mediator
- Coordinator
- Counselor
- Salesperson
- Clarifier
- Summarizer
- Diplomat
- Team builder

The *scribe* needs to be able to understand the topics discussed and to sort out what is important and what is not. While this is not a speaking role, the scribe asks for clarifications as needed. The scribe should be a good note taker, and possess the ability to stay focused.

In addition to selecting the right workshop participants, perform the following five steps to prepare for an elicitation workshop:

1. Establish workshop goals and objectives.
2. Become familiar with the business environment.
3. Decide the format of the deliverable from the session (Will it be a text document? What modeling techniques will you use?).
4. Create the agenda. The general format for a meeting usually includes something similar to the following:
    - Purpose
    - Schedule Confirmation (Date, Time, Duration, and Location)
    - Participants
    - Topics to Be Discussed
    - Preparation Action Items
    - Follow-up Action Items
    - Attachments
5. Prepare materials.

Focus Groups

A focus group is similar to a workshop, but consists of less participants and focuses on more specific topics that can usually be addressed in an hour to ninety minutes. A *Focus group* is a qualitative research method that is used to explore and deepen the knowledge of certain aspects and topics. It is a good practice to assemble a group of typical users, users of a previous service, or even users of a similar service. We suggest that you gather their ideas on functional and non-functional characteristics for the planned service by facilitating focus groups. Focus groups are an efficient and proven instrument to raise employee opinions and awareness about ongoing projects and change initiatives.

Sometimes, this technique is a better option than a one-on-one interview because a few more people will help to get participants actively brainstorming. Also, users often cannot think of everything they need when asked on the spot, but they can recall more when they hear others' needs. Focus groups reduce the spotlight on individuals, helping to produce more interesting answers.

Before Using This Technique

- Schedule meeting with participants. For time-pressed stakeholders and business analysts, it may be a good practice to hold focus groups virtually through an online meeting tool like GoToMeeting.com. Determine whether the meeting should be held online.
- Decide on the objective of the meeting and come up with some questions that you can ask.
- Create and finalize the agenda.

Interviews

The purpose of an elicitation interview is to accurately and efficiently discover information that can be used as an input to requirements analysis and modeling. Interviews can require a significant amount of preparation and good communication management to ensure they go smoothly and provide the desired results. An elicitation interview comes down to one idea—building a great product begins with asking great questions. And even before you start asking questions, you need to decide on what it is you want to achieve during the session. For example, do you want to explore broad options, understand a specific business process, or learn all you can about how a customer interacts with a particular feature? These questions are the basis for an interview objective, which, if accurate, will generate a successful elicitation.

Before Using This Technique

1. **Define the objective of your interview.**
   This should be done before the scheduled interview. The analyst's goal should be to capture good requirements for the future, not the personal agenda of one stakeholder. An elicitation interview has at least one of the following objectives:

   - Record information to be used as input to requirements analysis and modeling.
   - Discover information from interviewee accurately and efficiently.
   - Reassure interviewee that his/her understanding of the topic has been explored, listened to, and valued.

2. **Schedule the interview.**
   At this point, you're just making sure you have an appointment reserved on the stakeholder's calendar; you're not planning the agenda. Schedule the interview far enough in advance to let stakeholders know the analyst is aware that they have jobs to do and that the analyst has a high level of respect for their time commitments.

3. **Ensure you have an understanding of the stakeholder(s) and his/her role.** This involves completing *Task 3.2 Conduct Stakeholder Analysis*. Get an understanding of their personalities, as well as any personal or professional biases. This information will be useful as you plan and conduct the interview.

4. **Determine how notes will be taken.** For example, will they be taken manually, via audio or video? Also, who will be transcribing the notes?

5. **Generate a list of possible questions.** There are two common approaches to interviews:

   1. **Open-ended.** The interviewer starts with a list of key questions and then expands on them based on the interviewee's answers. These should flow from more general items to more specific, keeping in mind that the list should not be used as a script for the interview but as a guideline.

   2. **Survey.** The interviewer pre-defines the questions to be asked and goes through the list with the interviewee. This approach is very similar to a written survey; however, there is a chance to expand on answers that may not be clear. In terms of what kinds of questions you should ask, you can categorize topics based on the broadness of the expected response. For suggestions, see *Asking the Right Questions* on page 19.

6. **Finalize the agenda.**

Observation

Karl Wiegers, an industry guru, calls this type of elicitation "a day in the life study." In the *observation* method, an analyst observes users (or prospective users) perform their work with an existing service. This is a great technique to validate the data that you have collected from previous elicitation sessions, and, in addition, it can serve as a stimulus for generating new interview topics and questions, catching errors in the existing system, and recognizing possible changes to the service that would improve workflow. The advantage that the observation method has over other methods is its provision of a comprehensive view of the actual work that gets done, rather than just trusting users to describe the steps they take to perform a particular duty. When the BA is observing users, s/he has three goals in mind:

- Answer questions that need corroboration, comparison, or confirmation.
- Collect what is ordinary/what people do on a day-to-day basis, with the aim of making the implicit explicit.
- Study the context of work and watch work actually being done.

Before Using This Technique

1. Schedule a session with a user.
2. Get as much of an understanding as you can of the user's role and responsibilities prior to the session.
3. Determine whether you'll be relying on handwritten notes, or whether you'll be using audio/video recordings to document the observation session.

Surveys

Surveys are a great technique to use if you need to collect a lot of information from many potential users in a small window of time. They help the stakeholder stay focused on the topic at hand. Well-designed *surveys*, or questionnaires, are a viable means of familiarizing yourself with user needs. Surveys are often used in conjunction with other elicitation methods. The business analyst often holds a focus group or requirements workshop before or after releasing an employee survey. Surveys are also a useful technique for quickly gathering data from a large group of participants that is difficult to get together in person, and they are an inexpensive way to gather objective data from stakeholders.

Before Using This Technique

1. Determine survey participants. Try to pick respondents that you think are capable of explicitly expressing their knowledge and that are committed to investing time and effort into elicitation activities.

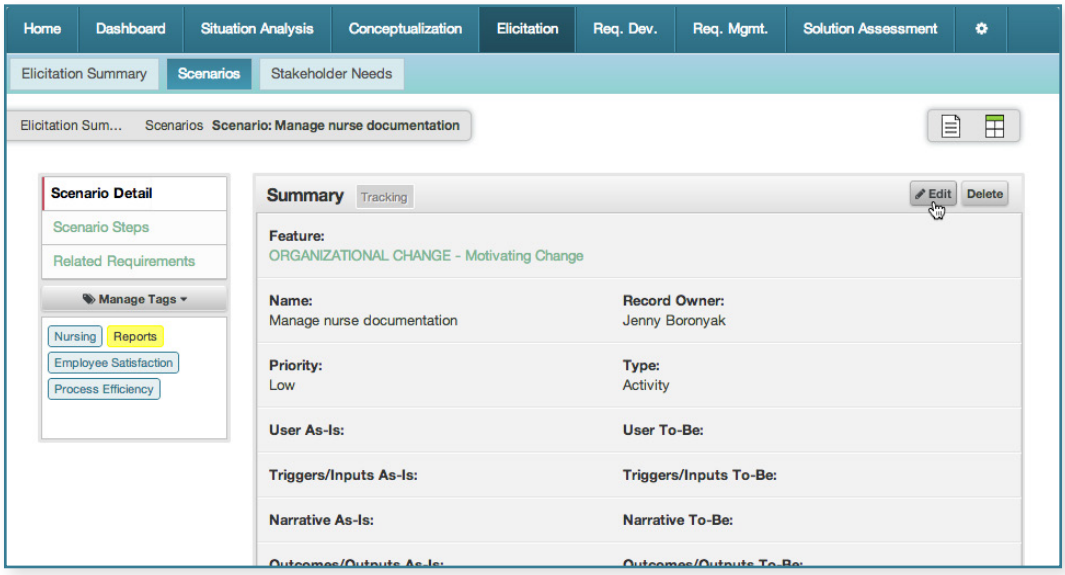
2. Determine the questions to be asked (Will they be open-ended, rated on a scale, or multiple choice?). Make sure the survey isn't too long that your stakeholders won't want to take it. The best approach is to pick five to ten key questions and ask them in a simple language.

3. Determine how surveys are to be returned and who will compile the results. Many organizations have turned to automating surveys by using online survey tools like SurveyMonkey.com.

4. Determine the Elicitation Deliverable.

It is best practice to document the results from elicitation events immediately after they occur. Regardless of the elicitation techniques you use, they will all result in information that must be documented for future use. With Enfocus Requirements Suite™, it's easy to create one complete elicitation deliverable that consists of all stakeholder needs gathered in all of the elicitation events related to the project. RequirementPro™ provides two features that aid in this process: Scenarios and Stakeholder Needs Patterns.

If the project team opts to gather Scenarios, the best practice is to first determine which ones need to be requested and document them in RequirementPro™. Then, assign each Scenario to a specific user that would know how to fill out the details of that record.



With a Portfolio Performance Subscription, you also have the option of documenting needs with Stakeholder Needs Patterns, which are step-by-step guides to documenting specific types of needs. Both RequirementPro™ and StakeholderPortal™ users have the ability to document their own stakeholder needs using this feature. If using Stakeholder Needs Patterns, the BA may either ask stakeholders to enter their needs personally, or enter the needs on their behalf subsequent to an elicitation event.

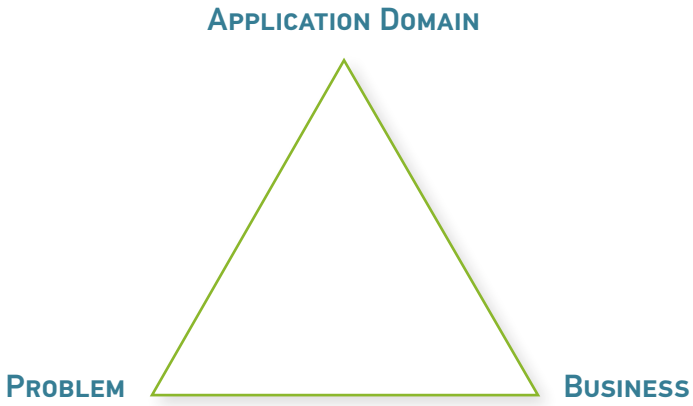
5. Determine the Schedule for Elicitation.

After agreeing on a set of elicitation techniques, the next step is to finalize the schedule for *all* elicitation events. Ensure all required participants are aware of their events and can attend. A good practice is to create one comprehensive schedule and distribute to all project contributors and stakeholders.

Don't forget to deal with stakeholders that cannot participate. The project team still needs their input. Some stakeholders may be unavailable or refuse to participate. It is important the business analyst does not ignore these stakeholders. There must be a plan in place for dealing with individuals who cannot participate in elicitation. Their input is just as valuable as that of the stakeholders who are able to participate.

6. Ensure Stakeholders Understand Business Requirements.

Before stakeholders can provide their needs, they need to have a complete understanding of three basic areas. The information that helps to create that understanding was documented in the two previous task groups, *2.0 Situation Analysis* and *3.0 Solution Conceptualization.*



- *Application Domain Understanding:* knowledge of the general area where the system is applied.
- *Problem Understanding:* details of the specific problem where the system will be applied.
- *Business Understanding:* how systems interact and contribute to overall business goals.

The easiest way to make sure stakeholders have the information that will help them understand the situation is to document everything in RequirementPro™ and provide stakeholders with access to that information via StakeholderPortal™. To provide the BA with their needs, stakeholders need to have an understanding of the solution. It is the responsibility of the BA to provide the stakeholders with all high-level information about the project, like the Problem Statement, Vision Statement, Objectives, Constraints, and Features, which are all available in Enfocus Requirements Suite™. Much of the information determined in the Business Case will be beneficial to stakeholders attempting to write their needs.

7. Determine Document Types to Be Gathered.

Determine how you're going to perform *Task 4.3 Gather and Analyze Documents*. In addition to the stakeholders involved in the project, one of the best sources of information for requirements development is documentation that already exists in the organization. Although these are often out of date, poorly written, wrong, etc., they are a good starting point for discovering requirements. The types of documents that you may need to gather can include the following:

- User documentation, like manuals and guides
- Development documents
- Requirement documents
- Internal memos
- Change histories

For more suggestions on types of documents to gather, refer to Chapter Three.

8. Determine Rule Books to Be Gathered.

This step can be performed once the project team has determined which areas of the business are impacted by the change brought on by the solution. To prepare for *Task 4.4 Gather Business Rules*, determine the following:

- Will you need to create new rule books?
- Will you need to retire any currently active rules or rule books?
- Will you need to revise any currently active rule books?
- Who will be on the authoring team for each rule book?

9. Determine How Terms Are to Be Documented.

Like Step 8, this step is performed once the team has information concerning the impacts on the organization. To prepare for *Task 4.5 Document Terminology*, determine the following:

- What new terms must be documented?
- What old term definitions must be retired?
- What existing terms must be edited?
- Who will be on the authoring team of the glossary?

10. Determine How Assumptions Are to Be Documented.

Based on the results of *Task 3.2 Conduct Stakeholder Analysis*, determine who will be good sources of information concerning the assumptions related to the project.

Elicitation Techniques Quick Reference

TECHNIQUES	DESCRIPTION	USE WHEN	ADVANTAGES	DISADVANTAGES
Direct Entry into StakeholderPortal™	Stakeholders directly enter needs into the system using Scenarios and Stakeholder Needs Patterns.	Anytime stakeholder needs are required, especially when stakeholders are geographically spread.	• Stakeholders provide needs when it is most convenient. • Analysts get feedback from stakeholders in various locations. • Documented conversations between stakeholders.	• No face-to-face contact. • May be difficult to get certain stakeholders to participate, like executives.
Interviews	One-on-one conversation between the analyst and stakeholder.	Few experts exist; many experts have the same knowledge base; or, follow-up questions are required.	• Easy to schedule because they involve two or three people. • More informal than other techniques, which may work best with some stakeholders. • Flexible and can go off into unplanned directions, allowing for more exploration of different topics. • Allow for more proving into details than other techniques.	• The customer feels less committed to an interview. • It is more difficult to control interruptions like phone calls or other people coming into the room. • Time consuming.
Observations	The analyst personally observes the stakeholder while s/he performs everyday tasks.	Must have an understanding of what the user of an existing system does on a daily basis.	• Allows analysts to see what users actually do, not what they think they do. • Multiple users can be observed at once to examine differences in individual practices. • Removes the need to take the user away from the work environment. • Less abstract than interviewing.	• Often difficult to discern reasons for behavior. The analyst may not understand why the user does what he does. • People act differently when observed. • Time may be wasted watching users do activities outside of the project boundaries. • Recording/ documentation may be difficult.

TECHNIQUES	DESCRIPTION	USE WHEN	ADVANTAGES	DISADVANTAGES
Studying Existing Systems	Review of current systems and related information.	As a starting point for gathering information before other elicitation activities.	• Limits stakeholder involvement. • Gives a jumpstart to elicitation with a lot of information. • May point out valid requirements built in the current system that the users would not think of mentioning.	• May not have updated information. • May not reflect the way business is done. • May be too time consuming for little ROI.
Surveys	A document distributed to stakeholders consisting of open- or close-ended questions.	Must elicit information from a large amount of stakeholders.	• Can reach a large population. • Can focus users on specific topics. • Can be used to prioritize requirements.	• Can be difficult to probe into negative responses. • May be difficult to get surveys returned. • No immediate feedback.
Elicitation Workshops	A formal, facilitated meeting to go over system and stakeholder needs.	Large group of stakeholders must reach consensus.	• User involvement and ownership. • All participants hear the same message. • Disagreements are solved on the spot.	• Can generate conflict. • Hard to facilitate. • Takes a high level of organizational commitment.
Focus Groups	A less formal meeting than a workshop that includes fewer participants.	Require a consensus among representative stakeholders with varying knowledge of the system.	• Generate a lot of ideas in a short time frame. • Saves time and cost compared to one-on-one interviews. • Allows participants to view other opinions and build on them.	• Can deteriorate because participants are not decision makers. • Trust issues within the group may lead to self-censorship. • Can be difficult to get everyone together at one time.

Asking the Right Questions

Regardless of what elicitation techniques you end up using in your project, you're going to have to ask questions to stakeholders. It is a good practice to brainstorm as many questions before actually performing elicitation techniques. The questions provided in this section can be applied to just about any method of elicitation.

As you think about the questions you will ask, make sure you're asking the *right* questions, which means you know why you're asking a question and what kind of response you expect to hear. Are you seeking facts or an opinion? Do you need the interviewee to explain something? What's the appropriate level of detail in the interviewee's responses? When preparing the list of questions, arrange them in a logical order, and keep in mind how much time will be spent on each issue. Write the list of questions in an order that maintains the direction of the meeting.

According to Hans Jonasson in *Determining Project Requirements: Mastering the BABOK and the CBAP Exam*, there are four types of questions used by interviewers. Interviews generally follow a pattern that begins with open-ended questions and finishes with validation. Remember during the interview to focus on "what" must be done, not "how." It is a good practice to keep in mind these types of questions as you perform other elicitation techniques, such as focus groups or workshops.

1. **Open-ended:** These questions allow stakeholders to open up and be descriptive about the work they do. Interviewees have the opportunity to introduce topics that the interviewer may not be aware of.
    - What are some of the difficulties with the current process?
    - What would an ideal system look like to you?
2. **Close-ended:** This type of question requires brief answers that are intended to keep the stakeholder focused and to let them know what type of information is needed. Sometimes, a simple yes or no will suffice. Close-ended questions can be prepared ahead of time or come from answers to questions asked earlier.
    - How many customers in a day will perform this task?
    - Of these customers, how many have this specific problem?

Depending on the type of response you want, it may be a good idea to change close-ended questions to open-ended questions by adding the words "what" or "why" to the beginning.

3. **Probing:** These questions are used to clarify statements. Probing questions help to minimize the amount of undocumented assumptions. Examples of good probing questions when the stakeholder says that the system must be available during regular working hours:
    - Do all locations have the same open hours?
    - Do you plan to expand into oversea markets?

4. **Validating:** During a lengthy interview, it is a good practice to pause every once in a while to ensure there is agreement on what has been discussed so far.
    - Does this flowchart accurately reflect the current process?

Roxanne E. Miller's book, *The Quest for Software Requirements*, is filled with suggested questions that will aid in the solution requirements development process. Enfocus highly suggests referring to this resource for advice on asking more specific and/or technical questions in an interview. Many of the general questions listed here come from her book.

- Context-free Questions
    - → What problems would this solution solve?
    - → What problems could this solution create?
    - → What obstacles or roadblocks are there to the success of this project?
    - → What other changes are occurring in the organization that may impact this project?
    - → How long has this concept been considered? What were its previous innovations?
    - → How will you use this feature?
- User-focused Questions
    - → Describe your responsibilities. Is there anything you would like to change about them?
    - → What business decisions (or business rules) do you make in your job?
    - → With whom do you interact?
    - → What information do you use in your job?
    - → What systems do you use in your job?
    - → What problems currently inhibit you doing your job?
    - → What training did you receive?
    - → How do you measure success in your job?
- Process-related Questions
    - → What processes are related to the solution?
    - → Who is involved with each process?
    - → What are the rules that each process executes?
    - → What data does each process need to be able to execute?
    - → How fast should the process be?
    - → How many transactions must each process be able to perform?
    - → Where will the process be used?
    - → Who is allowed to perform the process?
    - → How do we know the process is complete?
    - → Who delivers the inputs for the process?
    - → Who will receive the outputs for the process?

- Probing Questions/Prompts
  - → Tell me more about that.
  - → Explain what you mean by that.
  - → Describe what happens next.
  - → What do you think about this?
  - → How do you feel about it?
  - → Where could the workflow be improved?
  - → What concerns do you have?
- Questions that Assess Interview Progress
  - → Do my questions seem relevant?
  - → Are you the right person to answer these questions?
  - → Is there anything else I should be asking you?
  - → Are there important topics I haven't mentioned?
  - → Is there anything you would like to ask me?
- Questions that Keep the Interview on Track
  - → So that we can keep in our allotted time, may I move on to the next question?
  - → Is this topic more important than others, or should we continue to the next topic?
  - → Did we capture the key points on this topic? May we move forward?
  - → We've spent about ten minutes on this topic. Should we continue this discussion or continue on to the next topic?

Task Completion Checklist

To ensure the project team has adequately prepared for elicitation tasks, quickly refer to the following checklist:

	✓
Date/time/place confirmed for elicitation events.	
Elicitation event agendas prepared.	
Pre-event review deliverables identified and distributed to the participants.	
If video/audio recording devices or note-takers will be present at the event, prior consent has been obtained from the participants.	
Lists of questions and discussion topics prepared.	
If using Scenarios, determined Scenarios to be requested.	
If using Scenarios, assigned each Scenario to a user.	

4.2 Chapter Two
Gather Stakeholder Needs

Introduction

Elicitation is the process through which customers and users of a software system discover, reveal, articulate, and help the business analyst understand their requirements. In the opinion of many professionals, this is the most critical stage in requirements development. Elicitation serves as a foundation and prerequisite for other task groups and business analysis activities. If elicitation is performed incorrectly, the developers will end up building off of weak or incorrect requirements, resulting in unsuccessful solutions that require costly rework.

The business analyst's goal at this stage in the project is to develop an understanding of the user requirements that can be communicated to the development team. But this can be difficult when stakeholders do not understand how software design and development works and cannot specify their own software requirements in a way that works for developers. Because, then the software developers usually do not understand the problems and needs of stakeholders well enough to specify the requirements, ending in a cycle of miscommunication that occurs too often in software development projects. For this reason, the BA carefully elicits needs from stakeholders in *Task 4.2 Gather Stakeholder Needs*, and then reviews the needs to create requirement specifications that are understandable to both the stakeholders and to the solution team that will be building the solution.

Core Concepts

Scenarios

In Enfocus Requirements Suite™, a Scenario is one of two ways in which stakeholder needs can be documented (the other being Stakeholder Need Patterns). Scenarios are records that focus on the user and the specific actions they need to perform to accomplish a goal. There are three types of scenarios:

- *Activity*: a description of an activity that is or will be commonly performed.
- *Interaction:* a description of an interaction between two entities, either individuals or software interfaces.
- *Problem:* a description of a problem that has occurred.

Each Scenario record consists of information about its trigger, narrative, outcome, and the steps that lead to that outcome.

A major benefit of a scenario is its ability to be used and grasped by people without a specialized background. Therefore, you are able to utilize scenarios while completing design activities that involve user participation. As such, these scenarios are beneficial when used to convey an interaction from the perspective of an end user. This provides a common language that connects customer problems and technical solutions. Using scenarios also allows the more technical aspects of an issue to remain, appropriately, with the designers.

An example of a scenario can be found on page 31.

Stakeholder Needs Patterns

Besides Scenarios, the other way to document stakeholder needs in Enfocus Requirements Suite™ is through the use of Stakeholder Needs Patterns. A Stakeholder Need Pattern is a guide to writing a particular type of need. Patterns are created to explain how to approach each type of need, what to enter into each field, and what to worry about. Need patterns help users write better stakeholder needs and solution requirements by pointing out the details that must be focused on when it comes time to develop requirements.

An example of a Stakeholder Need Pattern can be found on page 25.

Process

How to Perform Each Elicitation Technique:

Analysis of Existing Systems

As you analyze the existing system, identify any necessary system interfaces. If there is a current system in place and if the business process is not going to change much, most interfaces can easily be discovered by analyzing where data is coming from and what triggers it, as well as identifying where information is being sent to other areas. As you analyze existing systems, your goal is to get an understanding of...

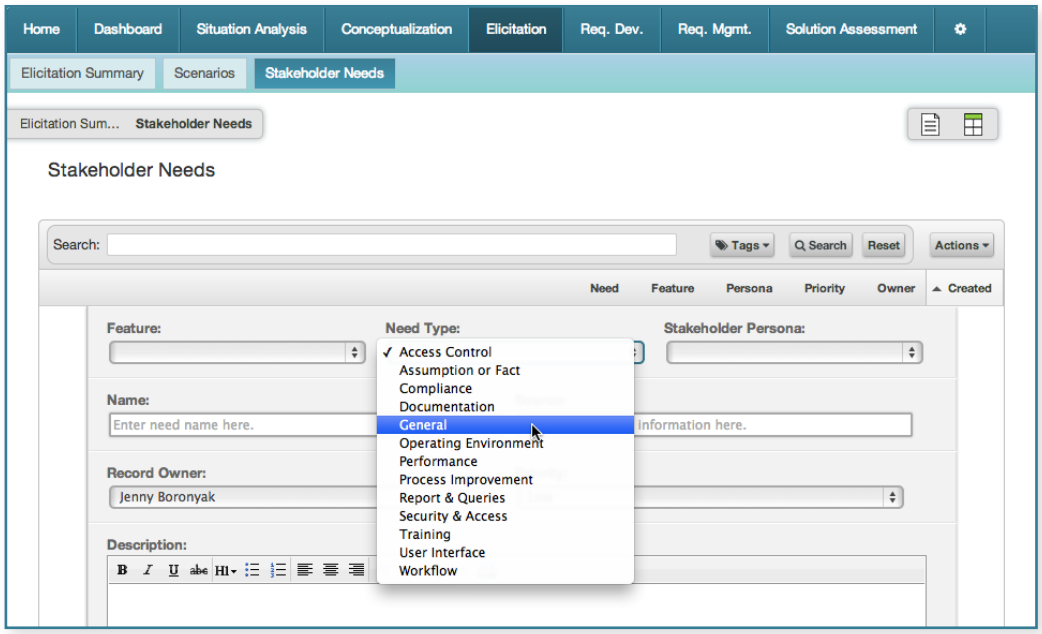
- What is used, not used, or missing.
- What works well, what doesn't work.
- How the system is used (with frequency and importance), how it was supposed to be used, and how users would like to use it.

Direct Entry via StakeholderPortal™

This elicitation technique is simple to perform from the BA's perspective. Once you have identified the stakeholders from which you must gather needs, reach out to those stakeholders and request that they provide their needs using StakeholderPortal™. The easiest way to do this is by publishing an announcement which will appear on the Project Dashboard and in the stakeholders' inbox. In the announcement, let the stakeholders know the following:

- Before they provide their needs, let them know they need to review the Project Overview, including Project Vision and Business Case, as well as the solution's Features.
- Inform stakeholders of the type of records you're looking for, whether they're Scenarios or Stakeholder Needs Patterns. Remember that, depending on your subscription, you may only have the option to request that stakeholders provide Scenarios.
- Give the stakeholders a deadline and let them know what the next step is once you have their needs. Without understanding the importance of elicitation and where the project is on the roadmap, stakeholders will often ignore your request for information.

Once stakeholders are aware that their needs are being gathered, individuals will use either Scenarios or Stakeholder Needs Patterns to enter their needs into StakeholderPortal™.



Elicitation Workshops

The facilitator of an elicitation workshop usually follows an agenda similar to the following:

1. Deliver opening speech.
2. Review agenda and objectives.
3. Establish ground rules. Examples include:
    - Silence cell phones.
    - No personal attacks.
    - Return on time after breaks.
    - Limit discussion on any one item to 10 minutes.
4. Get buy-in to process. This means everyone reaches agreement on the elicitation process.
5. Conduct session.
6. Document issues.

As you conduct the workshop, be aware of what state the team is in. Are participants working well together, or are they fighting? Be careful to make sure that you appropriately react to the participants' attitude. Periodically throughout the meeting, assess whether the goals and objectives outlined at the beginning of the workshop are being met. Remember that even though a topic might be relevant to the project, it may not be relevant to the workshop objectives.

One common challenge in workshops is timekeeping. When it's time to end a discussion on a certain topic but the team wants to keep talking about it, try handling the situation in one of three ways:

- Assign a subteam to take the discussion offline and come back with a recommendation.
- Defer other items on the agenda to extend the current discussion.
- End the discussion and vote on a decision.

Focus Groups

The process for facilitating focus groups:

1. Open with introductions and a description of the objectives of the focus group.
2. Explain the means of documenting the session, whether it's in writing or via video/audio.
3. Review the agenda.
4. Conduct the session. During the session, remember:
    - Take time to carefully phrase your questions.
    - A good practice in focus groups is to present a question, allow the group a few minutes to record answers, and then facilitate a discussion around the answers.
    - After someone answers your question, reflect back a succinct summary of what you heard.
    - Ensure everyone in the meeting is participating. A good practice is to use a round-table approach.
5. Close the session. Ensure participants are aware that they will receive a copy of the report generated from their answers. Always remember to thank the stakeholders for their participation.

Interviews

The basic approach to interviewing stakeholders is to identify key people in the organization, interview each of them individually, and then analyze the results. This technique works well when few individuals are involved and little disagreement is expected. According to Roxanne E. Miller in her book *The Quest for Software Requirements*, you always want to achieve a STAR interview:

- Start with Style
- TQLR (apply the active listening strategy: Tune In, Question, Listen, and Review)
- Ask Questions
- Recap and Wrap-Up

Follow Miller's guidelines for a STAR interview:

Start with Style

- Begin with introductions that establish good rapport.
- Review the objective and goals.
- Take notes or record the interview.
- Stay on track.
- Pay attention. Remember tricks like using the person's name and maintaining good eye contact.

TQLR

- Tune In: Give the interviewee your full attention.
- Question: As you listen to the interviewee's responses, ask questions to test your comprehension of the topic. Examples include, "What is the purpose of what I am listening to?" and "What is new in what I'm hearing?"
- Listen: Listen for clues and/or phrases that help you predict what's to come.
- Review: Confirm what you understand, which is not the same as repeating what you heard.

Ask Questions

- Use a prepared list of questions to navigate through the interview. Leave room in the interview for a follow-up questions.

Recap and Wrap-Up

- This means you leave the interviewee informed of the next steps in the process. Plan time for a recap at the end of the interview.

1. Summarize the information elicited.
2. Discuss and assign unanswered questions.
3. Solicit and answer any questions about the interview.
4. Discuss the next steps in the requirements development process.
5. Evaluate the interview process.
6. Thank the interviewee for his/her time and participation.

Remember that stakeholders often focus on what they think is important from their point of view. They'll leave out some information because they assume that the analyst already knows it. Different stakeholders will make different assumptions. By verifying and comparing different stakeholder needs, the analyst can identify conflicting assumptions and needs.

Once you have completed an interview session, review and revise the notes taken during the session. The business analyst should do this shortly after the interview so that the details are not forgotten. As you review the notes, identify inconsistencies and ask follow up questions via email or other forms of casual correspondence. Then, enter the results of the session into RequirementPro™. Possible methods of doing this are to assign to a stakeholder the task of entering the discovered needs into StakeholderPortal™, or the business analyst can also enter the discovered needs via RequirementPro™.

Observation

The key to this technique is observing specialists in their *native environment*. Observation can either be performed actively, by involving interaction with the user, or passively, by using video recordings or two-way mirrors. Initially, observe silently. Otherwise, you may receive biased information. Eventually, you can ask the user to start explaining each step of the process as he or she performs the work.

Keep in mind the Hawthorne effect, which posits that observing users will change the way they do their jobs. Be aware of making the user uncomfortable, as this can skew results. There must be a strong emphasis on building trust with the individual being observed. For both parties, there should be a clear understanding of what is to be observed, why, and for what the data will be used.

Remember that the observation technique works very well in conjunction with other elicitation techniques. A good practice is to supplement what you learned from observation with follow-up questions. The observation will provide a good idea of what users need, and the interview helps to make your understanding of their needs more complete.

Surveys and Questionnaires

One benefit of surveys is that, once they have been written and distributed, there's not much else for the BA to do until it's time to gather the completed surveys. When using this technique, remember the following points:

- When you distribute the survey, try to stress the importance of it so that stakeholders make an effort to complete it with well thought out answers.
- If there is a concern that the surveys won't be returned on time (or in some instances, at all), get management involved and showing support by asking survey takers to make it a priority.

Documenting Needs in Enfocus Requirements Suite™

As elicitation events occur, the results must be documented to be utilized later in the project. The documentation format should be predetermined and agreed to by the stakeholders. Remember that different elicitation techniques lend themselves to different types of documentation. Depending on the activity, documentation during an event may require whiteboards, audio or video recording, or simply writing down results. Compiling these results and entering them into RequirementPro™ ensures stakeholder transparency and requirements traceability.

There are two possible methods for entering stakeholder needs into RequirementPro™ once they have been elicited. After all elicitation activities have been performed, use the method that works best with your organization's methodology to enter needs into Enfocus Requirements Suite™, or combine the methods that work best with your business. Possible methods include Scenarios and Stakeholder Needs Patterns (remember that patterns are only available with a Portfolio Performance Subscription). It is possible that the project may require both methods. The main difference between the two methods is that the project team will assign specific Scenarios to individuals, but when it comes to Needs Patterns, the BA will generally ask stakeholders to provide their needs using the pre-defined patterns provided with Enfocus Requirements Suite™.

Guidelines for Documenting Stakeholder Needs

Before your stakeholders provide their needs, whether in the form of Scenarios or Stakeholder Needs Patterns, provide them with the following advice to guide them in the process:

Ensure needs are free of implementation. They should state WHAT is needed, not HOW to provide it. Although stakeholders are great sources for information regarding what they need to be able to do, they are not very good at describing the best method of providing that ability.

Describe all necessary functions. The detail of stakeholder needs should be sufficient to meet project goals and objectives, since the needs will eventually be used to develop requirement specifications. All needs must map to individual requirements.

Include all performance needs. It is the responsibility of the business analyst to ensure all need categories have been covered, but s/he can't determine that without the help of stakeholders. Think about elements of timing, throughput, accuracy, and precision. These needs will provide the basis for non-functional requirements.

Needs must be realistic. Remember that the needs list is not a wish list. Any impossible needs must be removed from the list of stakeholder needs.

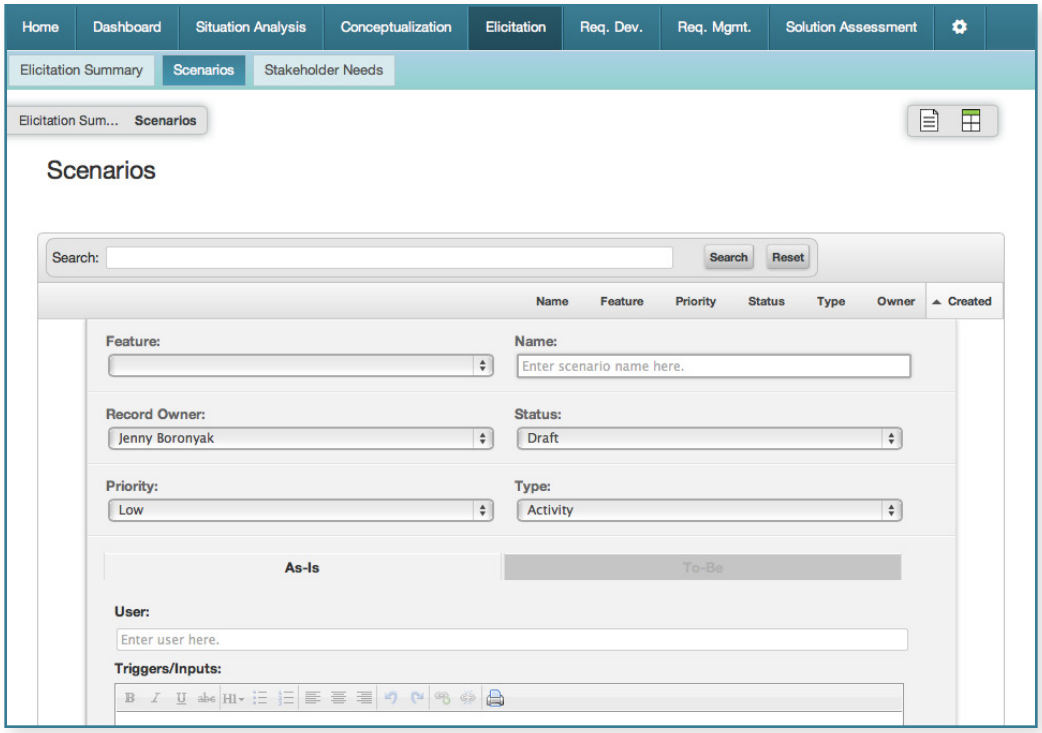
Can be tested and verified. Although you may not be able to determine the actual measures and acceptance criteria just yet, you must be able to determine whether measures can be set.

Needs must agree with business requirements. With the help of business stakeholders, the business analyst must analyze the needs to ensure they fit within the project scope.

Using Scenarios

A *Scenario* is a real-world narrative of the process that accomplishes a specific user goal. In comparison to Stakeholder Needs Patterns, Scenarios point to a clearer definition of the path a user will go on, as well as the tools the user will need to reach the end of that path. This method should be used to identify various kinds of flows and objects that are needed to accomplish a task by trying to understand how the user goes about solving problems. The purpose of a scenario is to get a clear picture of the user interaction with your system without including details of system actions or actual design. A good scenario will articulate the goal, process, and outcome contextually relevant to a type of user.

1. BA defines the set of Scenarios. The set must be agreed upon by the project team. To determine which Scenarios need to be developed to create requirements specifications, review and research the list of Features in RequirementPro™ to get an understanding of the tasks that will need to be performed. Also, talk to stakeholders to get an understanding of their daily work activities. Consider the following list of examples of common Scenario topics, created by the authors of *Business Analysis Techniques: 72 Essential Tools for Success:*
    - Common tasks and the responses to important business events
    - Situations involving a selection of users
    - Critical events that happen occasionally
    - Situations that are difficult to deal with
    - Situations where users are likely to make mistakes
    - Different working environments
    - As-Is and To-Be situations
    - How any new technology might be used
    - Boundaries between tasks and handoffs between users
    - Interleaving of tasks, showing their likely sequences

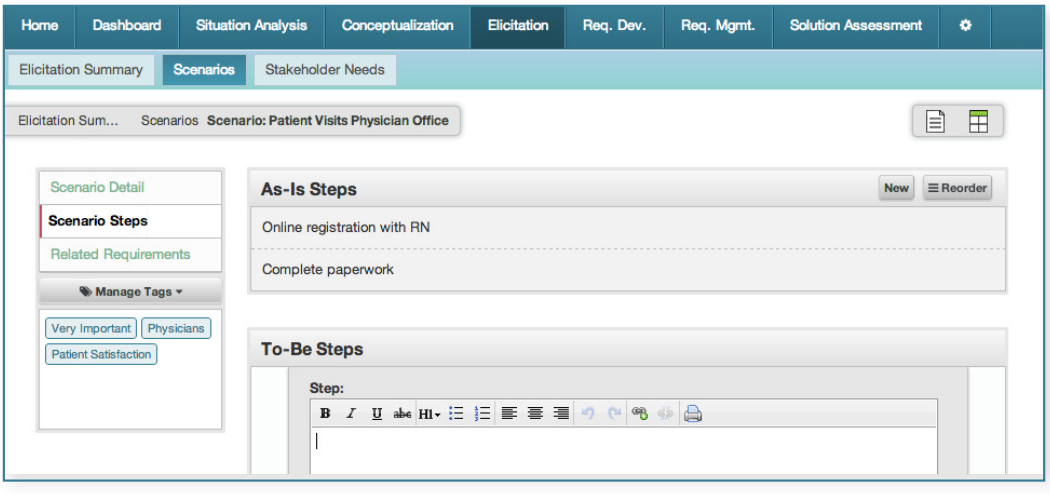


2. Enter high-level Scenario information into RequirementPro™. For each Scenario, the BA must enter the following:

   - *Feature*: From the drop-down menu, select the feature related to the Scenario.
   - *Name:* Provide an identifier for the record.
   - *Record Owner:* From the drop-down menu, select the project contributor or stakeholder to whom you will assign responsibility of completing the Scenario. This should be a specific user that has the knowledge to sufficiently complete the steps and provide required information related to the Scenario.
   - *Status:* This should remain "Draft" until the user has completed the Scenario. As the status of the record changes, change the status accordingly: Reviewed (by the BA), Mapped (to an existing solution requirement), Omitted (by the BA), or Withdrawn (by the stakeholder).

3. Let the Record Owner know the Scenario is available to be completed by assigning him/her an Action Item. The Record Owner will be responsible for entering the following information:

   - *Priority:* This will be based on the user's perception of the Scenario's priority.
   - *Type:* The type will be one of three options:
     - → Activity: A description of an activity that is or will be commonly performed.
     - → Interaction: A description of an interaction between two entities, either individuals or interfaces.
     - → Problem: A description of a problem that has occurred.
   - *General As-Is and To-Be Information*: In these two sections, the information is similar, with one big difference: the As-Is information will describe how the activity, interaction, or problem is currently performed, and the To-Be information will describe how an activity or interaction is to be performed in the future. These two sections include general information such as...
     - → *User*: The primary individual performing the Scenario; the perspective from which the Scenario is told.
     - → *Trigger/Inputs*: What causes the activity, interaction, or problem to occur.
     - → *Narrative*: A general description of what occurs.
     - → *Outcomes/Outputs*: What happens as a result of the activity, interaction, or problem.
   - *As-Is and To-Be Steps*: Once all of the prerequisite information has been entered and saved in the application, the Record Owner will complete the Scenario by describing the individual steps that are performed. These steps will be similar to the Narrative, but the steps tend to be clearer, and they are easily rearranged. When entering Scenario steps into StakeholderPortal™ or RequirementPro™, you do not have to worry about sequencing them correctly on the first try. Our application provides you with the ability to re-sequence the steps in a scenario at any time so that the scenario always reflects the current or desired process.



Using Stakeholder Needs Patterns

Stakeholder Need Patterns are a feature that is provided with the Portfolio Performance Subscription of Enfocus Requirements Suite™. In addition to Scenarios, when documenting needs in StakeholderPortal™ or RequirementPro™, you have the option to use one of the thirteen patterns defined by the analysts at Enfocus Solutions. A *Stakeholder Need Pattern* is a guide to writing a particular type of need. The types of needs for which there are patterns provided in Enfocus Requirements Suite™ include...

- Access Control
- Assumption or Fact
- Compliance
- Documentation
- General
- Operating Environment
- Performance
- Process Improvement
- Report and Queries
- Security and Access
- Training
- User Interface
- Workflow

Every Stakeholder Need requires a few of the same pieces of information:

- **Stakeholder Persona:** Enter the name of the stakeholder persona that owns the need. Every need must come from the perspective of a stakeholder. This is not necessarily the individual that is entering the need into the application; needs may be entered by RequirementPro™ users on behalf of stakeholders if they do not have access to StakeholderPortal™.
- **Source:** Specify the source of the need. In this field, enter from where you collected the need's information. For example, is there a new regulation? Was this need decided in a meeting?
- **Name:** Each need should have its own unique, concise, and specific title.
- **Feature:** Every need must directly relate to a feature already defined by the project team. If your need does not address a feature, it does not belong in the current project.
- **Status:** Needs should be labeled with their statuses. In StakeholderPortal™, you have the option to select draft, reviewed, mapped, omitted, or withdrawn. The default setting is draft.
- **Priority:** The priority level is one of the most important pieces of information that you will provide to business analysts when you create a need. It provides business analysts with an idea of the features that need to be implemented first.
- **Description:** Since the need's name should be concise, in this field, provide all the detail necessary to make the need complete. Give time estimates if they are applicable.

You may find that none of the twelve patterns provided in Enfocus Requirements Suite™ match your needs. In that case, use the General Need pattern. A *General Need* describes a project aspect that is typically necessary, but does not fall under any other need category and is essential for system operation. The other twelve more specific patterns are listed and described below. Refer to this list if you are unsure about what information to enter into each field.

Access Control Need: Specifies who is able to use and view information within the system.

- **User Registration:** Specify the necessary process for adding a new user to the system. For example, do new users have to go to a webpage to sign up or are they invited?
- **User Authentication:** Specify how the users are to be recognized by the system. For example, do they enter a user ID and password? How long should the password be?
- **User Authorization:** Describe the process for authorizing new users. For example, do they get set up automatically when they sign up, or does a department head have to approve them?
- **Access Approval:** Specify the individual(s) who determine the roles people have and the data that each role has access to. For example, who can CRUD (create, read, update, and/or delete) records in the system?
- **Audit Controls:** Describe the need for keeping track of when users last signed on and how that information needs to be tracked.
- **Standard:** List any business rules or industry standards that must be followed in the report.

Assumption or Fact Need: Specifies an assumption or fact of how an IT service or component operates.

- **Source:** Every need requires source information, as listed previously. An Assumption or Fact Need could be collected in a meeting or from a conversation, but the assumption or fact probably comes from somewhere else. Cite from where the need originates.

Compliance Need: Helps ensure that any regulations from the business, government, or industry are documented and accounted for. A need that acts as a directive to address new internal procedures or Federal laws and regulations.

- **Version:** If the need is a result of an updated regulation or standard, state the name of the most recent version so that there is no confusion among project contributors.
- **Standard:** List any corporate policies or industry standards that must be followed. Enter the name of the policy.
- **Purpose:** State the reason why your organization must comply. For example, if it is not a law, are you contractually obligated?
- **Citation:** Provide the page number or reference number associated within the standard. Be very specific in your citation. Provide the paragraph number to guide people directly to the standard to which you refer.
- **Attachments:** It would be also be a good idea to attach a copy of the standard to the need record so that all contributors can easily refer to it.

Documentation Need: Outlines the purposes for creating a new document.

- **Purpose:** State the reason for the new documentation. For example, a support team for a new business might need a manual written to make their tasks easier to learn.
- **Content:** Determine and describe the content needed in the new documentation.
- **Format:** Describe exactly how the document needs to be formatted. Documents require different formats depending on the location of distribution.
- **Standard:** List any corporate policies or industry standards that must be followed in the document.
- **Availability:** State where the document is to be available. Will it be available on a web page or does a user have to download it?
- **Language:** Specify the language of the document. In large companies that span continents, this might be an important field.
- **Distribution:** Describe the method of distribution, which will have an effect on the format of the document. For example, is the document intended to be published on a website or printed in handouts? Also, state to whom it is being distributed. For example, is it going to be delivered to managers only or every employee?
- **Confidentiality:** State whether the document should be distributed to all external customers or internal users only.

Operating Environment Need: Describes the environment in which the solution will operate.

- Variants: List any possible exceptions. For example, if you are creating a product that must operate in high heat, would there be any situations where it also has to be able to operate in a colder environment?

Performance Need: Documents different aspects of performance, which is a measure of what is achieved or delivered by a system, person, team, process, or IT Service.

- **Performance Type**: Describe the characteristics that define the performance. For example, should the system be able to perform a function in a certain amount of time, or should it be able to handle a high number of transactions?

Process Improvement Need: Describes the required changes to existing processes.

- **Process Impact:** Specify the processes impacted by the change.
- **Current Problems:** Describe the problems with the way the process is currently performed.
- **Recommendations:** Generally describe the improvements to the process.
- **Changes in Inputs:** If any, describe the changes to the process inputs.
- **Changes in Outputs:** If any, describe the changes to the process outputs.
- **Changes in Controls:** If any, describe the changes to internal controls.
- **Changes in Resources:** If any, describe the necessary changes to the organization's resources.
- **Changes in Workflow:** Describe the impact of the process improvement on workflow.

- **Current Performance Measures:** Document the current metrics of the related process.
- **Target Performance Measures:** Specify the desired metrics once the improvement is in place.
- **Expected Benefits:** Describe what this process improvement will do for the organization.

Report and Queries Need: Describes the outputs necessary for a successful solution.

- **Purpose:** State the reason the report is needed.
- **Report Content:** Provide an extensive description of the data elements that must be in the report. For example, if a report of all employee information is necessary, the report will require names, addresses, phone numbers, identification numbers, etc.
- **Format:** Describe exactly how the report needs to be formatted. Reports require different formats depending on the location of distribution. For example, will it be displayed online or printed? After answering that question, should it be portrait, or landscape?
- **Standard:** List any business rules or industry standards that must be followed in the report.
- **Language:** Specify the language of the document. (For example, English, Spanish, Portuguese, etc.) In large companies that span continents, this might be an important field.
- **Sort Sequence:** Specify how the need will be sorted. For example, should it be alphabetically or by invoice number?
- **Selection Criteria:** Specify whether there are any criteria needed for limiting data. For example, should the report just pertain to your region or just your stakeholder persona perspective?
- **Security and Privacy:** Specify whether there are any security or privacy concerns with the report.
- **Distribution:** Describe the method of distribution, which will have an effect on the format of the document. For example, will the document be published on a website? Printed in handouts?
- **Frequency:** State how often the report is needed. For example, will it be only on demand? Will it be scheduled monthly?
- **Target Audience:** Describe the audience that will be reading the report. For example, is it intended for managers? Transaction users?

Security and Access Need: Allows the user to document privacy issues that can affect the delivery of a service.

- **Threat:** If the need is a response to a possible threat, describe the indicators or warnings of that probable trouble.
- **Internal Controls:** Describe the internal controls that will ensure the organization's resources are protected.
- **Data Security:** State whether there is confidential data in the system and how to protect it. For example, how do you secure information about employee salaries and social security numbers?
- **Network Security:** Describe any necessary network security such as encryption.
- **Transaction Security:** State who is authorized to perform a transaction.

- **Standard:** List any business rules or industry standards that must be followed in the area of security and compliance that relate to this need.

Training Need: Allows the user to document aspects and constraints for future training.

- **Number of Users:** Provide the number of users that are going to need training.
- **Delivery Mode:** Name and describe the best method for delivering training. Will it be in the form of online videos? Or will there be a workshop held?
- **Constraints:** It is important to note any budget, time, or technical constraints influencing the ability to train users.
- **Training Needs:** List and describe the areas in which users will need training to be successful with the new solution.
- **User Availability:** Determine whether the team can assume people will be available for training during work hours or if the training will have to be scheduled at another time. For example, will it have to occur during the weekend?

User Interface Need: Allows the user to document what improvements or modifications need to be made to the user interface.

- **General Characteristics:** State the defining aspects of the required general characteristics of the new user interface.
- **Standard:** List any business rules or industry standards that must be followed in the report.
- **Navigational Flow:** Describe the flow of the required user interface. You should list the best and most logical sequence of actions.
- **Presentation**: Describe how the user interface should be presented to the user.
- **Dashboard and Graphics:** Describe the need for dashboards or graphics. If there is a dashboard, describe what information and widgets need to be displayed.

Workflow Need: Helps document workflow process and the changes that need to be made to it.

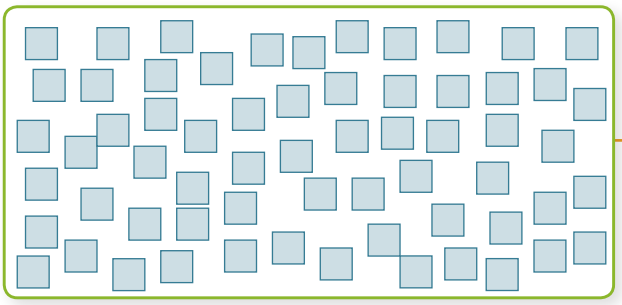
- **Process Owner:** Each workflow should have a process owner. Make sure to name the owner of the related process.
- **Impacted:** Name the stakeholder personas that will be affected by this need. Be descriptive and list the impact for each stakeholder persona, as well.
- **Interaction:** Describe the type of interaction anticipated between the process and the parties that are involved with it.
- **Access Approval:** Specify that a particular action must be approved before it takes place.
- **Audit Controls:** Describe the internal controls that are needed to address audit concerns.
- **Standard:** List any business rules or industry standards that must be followed.

Visualization and Elaboration

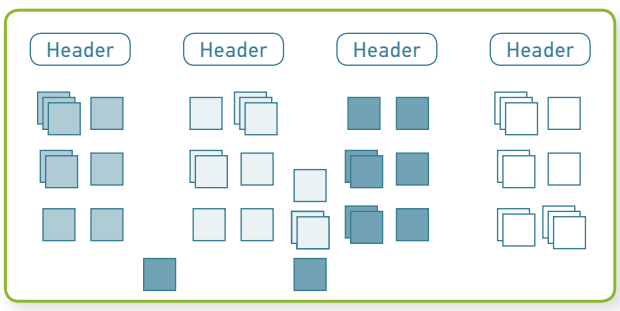
Affinity Diagram

An affinity diagram is a useful technique when an elicitation event involves multiple people brainstorming together. Affinity diagrams are used to organize any type of information, and they are useful for sifting through large volumes of data. Typically, affinity diagrams are used to brainstorm information, and then that information is organized to discover even more information. Depending on the project, it may be a good idea to use an affinity diagram as a brainstorming technique in some of your elicitation events.

1. Generate ideas in discussion.
2. Display ideas on whiteboard, computer screen, etc. First, the whiteboard will look something like this:



3. Sort ideas into groups.
4. Create a header for each group that captures the essential link among the ideas contained in the group.
5. Document completed diagram.



Storyboarding

A storyboard is a technique consisting of a series of diagrams or screen sketches showing navigation routes through a task or a series of screenshots. Storyboards are helpful to illustrate the structure and navigation of a system, and are often used to obtain feedback from interested parties.

This technique assists in understanding the big picture and breaking it down into smaller components that are easier to focus on. While storyboards are similar to prototypes, the advantage here is storyboarding can be done quickly with a pencil and paper and without the use of technology. This means that it is a great technique regardless of the participants' technical expertise. However, storyboards are only useful in certain projects. Assess whether it would be a good technique to use for your project depending on the complexity of the business problem, the budget, and the resources. The benefits of storyboarding are that it...

- Provides an overview of a system, including the manual and automated parts.
- Demonstrates the functionality of various storyboard elements.
- Demonstrates the navigation sequence.
- Helps identify where more information or analysis is required.

The BA may want to use this technique in a group discussion about a solution where there is significant interaction between people and systems. Before creating the storyboard, decide in advance how much detail is required. To create a storyboard, sketch each screen of the particular scenario(s) being described. Often, the original sketch is created on a whiteboard with Post-it notes before developing a formal process model or prototype.

Prototyping

According to James Martin, prototyping is a technique for building a quick and rough version of a desired system or parts of that system. Prototypes supplement discussions by providing a visualization of the end product. The purpose of prototypes is to ensure effective communication between stakeholders and project contributors.

Prototyping often involves storyboarding first. Before building a prototype, the project team must agree on the level of scope and functionality. Decide whether the prototype is to be broad in scope (horizontal), or if it is to focus on a smaller portion of the system (vertical).

Prototypes will vary according to the project; some won't require them. In some projects, the prototype becomes the product (evolutionary); in others, they're something called "throwaway prototypes." A throwaway prototype is useful when its purpose is solely to facilitate communication. On the other hand, evolutionary prototypes provide the advantage of not starting over. If you know the prototype will be used for production, prepare for it and put more effort in it than you would a throwaway prototype.

Task Completion Checklist

This task is pretty straightforward. As a reminder, make sure the following activities are performed to complete *Task 4.2 Gather Stakeholder Needs*.

	✓
Elicitation sessions completed.	
Notes from elicitation sessions reviewed and transcribed.	
All relevant Scenarios documented.	
All relevant Stakeholder Needs Patterns documented.	

Related Content in RequirementCoach™

Practice Aid: Stakeholder Needs Template: This template allows you to describe stakeholders and their needs with various characteristics such as basic assumptions/facts, general needs, reports/queries needs, and training needs.

Analyst Brief: Stakeholder Collaboration: The 2.0 Way: The Web 2.0 underpinnings of StakeholderPortal™ allow stakeholders to engage and collaborate throughout the various stages of the project lifecycle.

4.3 Chapter Three
Gather & Analyze Documents

Introduction

While it is absolutely vital that the project team gathers stakeholder needs directly from stakeholders, there can also be other good sources of information available to the BA. Don't overlook the wealth of information provided to you in written documents. Gathering and analyzing documents related to the project will help the project team ensure they have a complete set of stakeholder needs by supplementing the information gathered using the elicitation techniques that were performed in *Task 4.2 Gather Stakeholder Needs*. Regardless of the size of your project, there are existing documents from which you can extract information that will later aid in requirements development.

Core Concept: Document Analysis

Who?

This task is facilitated by the BA and requires a high degree of collaboration with stakeholders. The analyst will need input from subject matter experts who can point the BA in the right direction to find the documentation that needs to be reviewed.

What?

Document analysis is a means for eliciting requirements of an existing system or process by studying available documentation and identifying relevant information. It helps to provide an understanding of the As-Is environment with its existing business rules, relevant entities, and attributes that must be included in the new system being built. This information supplements other elicitation techniques, and is often performed before or at the same time as other elicitation events.

Many requirements can be extracted from review and analysis of documents such as business process models, existing user documentation, or rules and regulations. It is important to note that, often, these documents are out-of-date, poorly written, or incorrect; however, they are still a good source of information. In every project, the BA should read all available relevant documentation, like:

- User documents
- Development documents
- Requirements documents
- Internal memos
- Change histories

For example, if you are replacing a system, your current system may be a great source to elicit information from. If you are developing a product, then your may want to look at product brochures of your competitors. If you are implementing an IT system, then look at the process documentation of the process it will support. You may also want to look at trouble tickets, existing documentation, work guidelines, and policies and procedures.

When?

When the BA...

- is unfamiliar with the area of the organization being addressed.
- needs an understanding of the current situation surrounding the problem being addressed by the solution.
- needs to supplement information gathered from other elicitation techniques.

Why?

Advantages

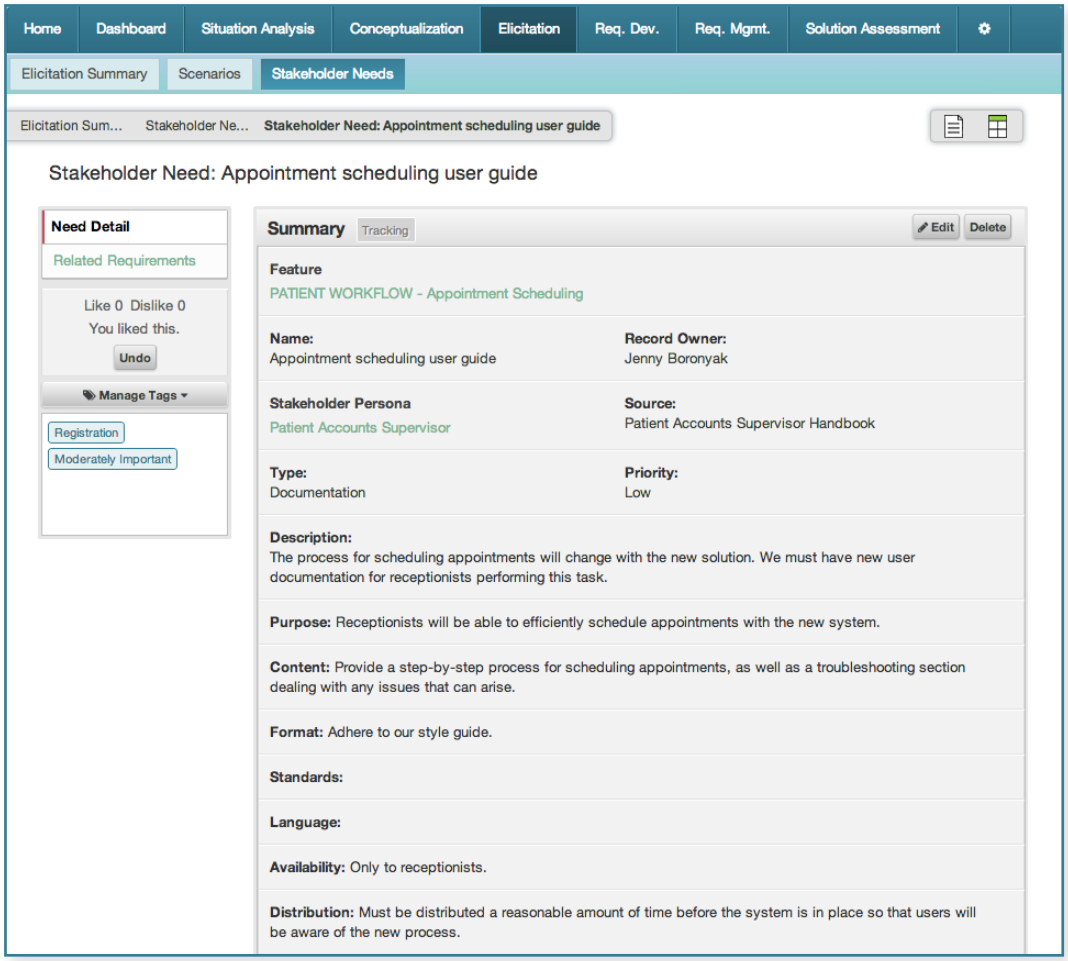
- The BA can leverage existing documentation to discover stakeholder needs.
- Document analysis a means to confirming stakeholder needs gathered from other elicitation techniques, such as interviews, observations, surveys, focus groups, workshops, or direct entries into StakeholderPortal™.
- There is no time commitment required from users or stakeholders.

Disadvantages

- The BA is limited to the As-Is perspective.
- Existing documents may not be up-to-date or valid.
- Day-to-day work often differs from documented procedures.
- Document analysis can often be a time-consuming process.

Process

1. Gather documents in Document Catalog. A list of possible documents is provided in the example of a Document Catalog on page 44.
2. Analyze documents. Identify and document relevant business details. Also, document any questions that you cannot answer, and mark them for a follow-up interview with stakeholders who are subject matter experts.
3. Review and confirm documented business details with relevant subject matter experts.
4. Document discovered stakeholder needs in RequirementPro™. Needs can be entered into RequirementPro™ (as well as StakeholderPortal™) via Scenarios or Stakeholder Needs patterns.



Stakeholder Needs patterns are available in the Portfolio Performance subscription. The Documentation Need pattern is displayed below. Depending on the type of document, you may not be required to fill in all of the available fields.

Document Catalog

Refer to the table below for an understanding of what goes into the document catalog, as well as a list of possible types of documents to gather.

CONTEXT	DOCUMENT	DOCUMENT OWNER	ASSIGNED BA	WHERE TO ATTACH
Governance	Funding Guidelines	Finance		
	Design Standards	IT		
	Data Governance	IT		
	Regulatory and Compliance	Legal		
	SOX/Internal Controls			
	Privacy/Security Regulations			
Project	Project Plan	Project Manager		
	Issue Information	Project Manager		
	Business Case	Project Sponsor		
Process	Process flows	Process Owner		
	Workflows	Process Owner		
	Forms	Process Owner		
	Standards	Process Owner		
	Performance Metrics Reports	Process Owner		
	Workarounds - Spreadsheets	Process Owner		
	Benchmark Data	Process Owner		

Document Catalog Cont.

CONTEXT	DOCUMENT	DOCUMENT OWNER	ASSIGNED BA	WHERE TO ATTACH
Product	Competitive Product Literature	Product Manager		
	Product Roadmap	Product Manager		
	Press Releases	Product Manager		
Data	Master Data	DBA		
	Data Volumes	DBA		
Knowledge	Communities of Practice Documentation	COP/COE Lead		
	Books and Other Reference Data	Business Analyst		
Organization	Staffing Numbers	HR		
	Position Classification Data	HR		
	Development Organization Chart	IT/HR		
	Business Organization Chart	Business Unit/HR		
	Staff Satisfaction Surveys	HR		

Document Catalog Cont.

CONTEXT	DOCUMENT	DOCUMENT OWNER	ASSIGNED BA	WHERE TO ATTACH
Technology	Existing System Documentation	IT		
	Transaction Volumes	IT		
	Automated Transactions	IT		
	Copies of Existing Screens	IT		
	Report Layouts	IT		
	Service Catalog	IT		
Problems	Trouble Tickets	Help Desk		
	Audit Findings	Internal Audit		
	Customer Complaints	Support Center		
Strategy	Balanced Scorecard Data	Project Sponsor		
	Metrics	Project Sponsor		
	Management Presentations	Powerpoint Presentation		

Task Completion Checklist

	✓
Gathered documents.	
Analyzed documents.	
Reviewed and confirmed details.	
Documented stakeholder needs.	

4.4 Chapter Four
Gather Business Rules

Introduction

Organizations will always have to gather business rules, whether they are self-mandated policies or governmental regulation and compliance. Industries with high levels of government oversight (like those involved with natural resources, food and agriculture, and energy production) will notice that they gather extensive collections of business rules. Often, business rules are constrained, exercised, and executed through personal and physical means, rather than through software, so most all business rules are actually gathered and established apart from software applications. Consider this business rule: "Hard hats must be worn when visiting a construction site." This is an example of a business rule that must be enforced through manual means. Despite examples of business rules that aren't managed through software, software applications are often needed to enforce them.

While there are tons of business rules, finding them can be hard. Business rules exist in old code, policy manuals, old user guides, and even in people's heads! To adequately gather business rules, a solid approach for gathering and documenting business rules is necessary. This guide will explain how to gather and document business rules with stakeholder collaboration.

Core Concepts

Business Rules Are Gathered And Developed By The Business

Because business rules define rules about a business, business rules should be gathered by business people, not IT people. In the past, IT has been known to gather business rules from the business and then convert those rules into complicated and technical language. This practice is troublesome because business rules need to be easily understandable, and therefore editable, by business people, not to mention the conflict of interest presented to IT.

A business rule policy gives business people more power, and rightly so; business rules operationalize broader business policies. For example, a policy such as "provide discounts to repeat customers" is further clarified by stating a more discrete, atomic business rule: "If a customer orders products totaling $1,000 or more in any calendar year, then offer a 10% discount on each non-sale product item." This business rule in turn is dependent on another, more fundamental business rule: "Each customer orders one or more products."

Business Rules are Best Organized in Rule Books By Functional Area

When conducting projects using the business rules approach, a plethora of rules are identified and defined. However, business rules are not useful if they are not organized in a way that allows organizations to find and group them together for review and analysis. Rules are most serviceable whenever organized into rule books by functional area and maintained separately from requirements. Rule books function as categorized containers for individual business rules. Each rule book has an owner in Enterprise Portfolio™, and the owner is responsible for maintaining the business rules contained within. Also, several rule books might exist for particular functions. For example, take human resources. In HR, rule books may exist for paid time off, time reporting, overtime pay, benefit eligibility, etc. Functional areas, or process categories that Enterprise Portfolio™ offers for categorizing rule books are as follows:

- Develop Vision and Strategy
- Develop and Manage Products and Services
- Market and Sell Products and Services
- Deliver Products and Services
- Manage Customer Service
- Develop and Manage Human Capital
- Develop Information Technology
- Manage Financial Resources
- Acquire, Construct, and Manage Property
- Manage Environmental Health and Safety
- Manage External Relationships
- Manage Knowledge, Improvement, and Change

Business Rules Are Organized By Function And Span Multiple Projects

Business rules can easily span multiple projects within a company, so they are organized by function for convenience. Rules can determine any number of things and decisions, so it makes sense that they could span multiple projects. Business rules dictate and encompass topics like how decisions are made, what information is valuable and valid, and what an organization's priorities are. Business rule function categories, or types as listed in Enterprise Portfolio™ will be explained in depth later on, and are as follows:

- Action Enablers
- Constraints
- Calculations
- Facts
- Term
- Data Tables

Write Business Rules to be Executable

Many organizations have a business rule engine (BRE) or business rule management system (BRMS) to capture their business rules. However, the business rules must be written in a certain format to be executable in the system. For example, IBM offers a BRMS that requires an If/Then format, among others. Such as, "If the customer's category is Silver and the customer's purchase is over $500, *then* change the customer's category to Gold." According to IBM, "In this form, the business logic can be packaged as an executable rule set and called from the application code as a single entity. Therefore changes to the business policy do not require changes to the application code."

Business rules only need to be written to be executable if your organization has an existing BRE or BRMS. Otherwise, the business analysts at Enfocus Solutions Inc. recommend writing business rules as according to RuleSpeak. RuleSpeak is a free online source that instructs business analysts and professionals on how to write easily understandable, consistent business rules.

Business Rules Should Be Discovered With Requirements Early On

Whenever business rules are not clearly documented early on in a project's process, it can result in a significant amount of costly project rework. Even when business rules are explicit, they may be vague and contradictory. Software teams often need formal guidance in uncovering, analyzing, and capturing business rules. Otherwise, developers simply make whatever assumptions are needed to write the code, building their assumptions into the software with little regard to the impact on the business. Inevitably, developers may guess wrong, and only during the latter phases of implementation will it be discovered that essential business rules have not been implemented. These late defects could have been avoided if the rules had been discovered early on in tandem with requirements development. The lack of explicit focus on capturing the business rules creates rework and other inefficiencies.

Requirements and business rules have a relationship in that requirements should reference the rules to ensure that the solution enforces the rules as planned. If an organization can trace a specific functional requirement back to the business rule from which it originated, it is easier to modify the system to comply with a change in that rule. Different functional areas may have different sets of rules, which need to be negotiated so a uniform set is applied.

Business Rules Are NOT Requirements

Business rules are not functional requirements; however, business rules may strongly influence functional requirements. Business rules exist outside the boundaries of software and, therefore, should be regarded as an enterprise level asset. However, business rules often require that specific functionality be implemented to ensure that the system enforces or complies with those rules. Requirements will reference business rules to ensure that the solution enforces the rules as planned.

Process

Every organization has business rules. In fact, some organizations are known to have tens of thousands. Article 1.1 of the *Business Rules Manifesto*, written collaboratively by the Business Rules Group, states that "Rules are a first-class citizen of the requirements world." This means that business rules should be externalized from all your documents and managed on their own right. Business rules are often embedded in business documents such as agreements, regulations, and marketing materials. They are also found in requirements documents such as use cases and business requirements documents. For a business rules project, these business rules should be specified independently from the other deliverables.

Business rules are often created without the context of specific software applications; they are owned by the business and should not be confused with software requirements. Regardless, it is easy to document needed requirements from documented business rules. After all, business rules dictate what properties a system should possess to conform to the rules.

In addition to being plain-language guidelines governing decisions, business rules should also describe policies for making decisions, formulae for calculations, business terms and definitions, and the vital facts and assumptions of how an organization is expected to operate. Simply stated, business rules are listed statements that inform organizations whether or not they are able to do something. The rules eventually provide the criteria and conditions you should know to make an informed decision. Business rules include corporate policies, government regulations, industry standards (such as accounting practices), and computational algorithms. Not every business rule is implemented in software.

Some general examples of business rules are:

- A past due account is an account that has not been paid in full after five business days of the payment due date.
- Past due accounts are subject to late fees, interest rate increases, and suspension of the customer's credit limit.
- Net sale is defined as the total sales price of an order before discounts, allowances, shipping, and other changes.
- When an order is initiated via the internet, sales taxes will be calculated by multiplying the net sale amount by the sales tax rate in effect in the state from which the order was placed.

The following steps detail the process for gathering business rules.

1. **Determine Impacted Domains.** Initially, the business analyst will determine which business domains will be affected by the project being undertaken. Knowing which areas of a business being affected by a project will point towards who should be involved in authoring business rules as well as what types of rule books will be made. Stakeholder analysis can be used to determine who stakeholders are in each domain and identify key people to talk to for collaboration. Organizational charts may also be evaluated for guidance.

2. **Determine Rule Books.** The second step of procuring business rules involves the business analyst and stakeholders determining what logical rules books would be for each domain. The business analyst is to make a list of possible rule books and have it reviewed by stakeholders for applicability. Rule books and their associated rules are well-suited to be maintained in a repository such as the one offered in Enterprise Portfolio™, as business rules often span multiple projects across an organization. Repositories are used to manage version control problems and ensure that anyone who needs to review a rule has access to the most recent version. Having just one repository helps to eliminate contradictions and duplicate business rules across multiple projects, making maintenance efforts across all projects considerably easier.

   It is recommended that a glossary is made at this point and that the glossary is made as a separate rule book defining terminology. *Task 4.5 Document Terminology* covers the aspects of documenting project and organizational terminology in depth.

3. **Assign Authoring Team.** Next, the business analyst will assign an authoring team for each rule book. Members of the authoring team should be from the business domain impacted by the rule book. Business rules can be found in documents or manuals, but often times business rules live in the heads of employees so choosing authoring teams from the domain the rule book will impact is effective.

4. **Business Rules Guidelines.** After assigning authoring teams, the business analyst is to assist the authoring teams in writing good business rules by hosting business rules writing workshops. By hosting workshops, the business analyst will be saving him or herself valuable time. There are good ways and bad ways to write business rules that authoring team members should be made aware of. Business rules writing workshops allow the business analyst to inform all business rule authors about the best practices for writing business rules. Authoring teams should not be thought of as writers or authors and should be given direction and assistance accordingly. Perhaps most important to consider is that business rules need to be written simply, using clear, succinct language. By design, business rules should not be complex; the more complex a rule is, the more you ruin one of its main appeals and beneficial capabilities. Clearly written business rules are rules that can be understood by everyone. If it takes multiple doctoral degrees to understand a business rule, then it has been written poorly and there is always the possibility of miscommunications and issues.

Below are some general guidelines for writing business rules.

- **Business oriented**—Business rules are stated in terms business people can understand. The rules must use terms that are meaningful, understood, and confirmed across the business domain. They should be expressed in such a way that business people can validate them for correctness.
- **Owned by the business**—Business rules are created and maintained by business people and not by IT. Article 9 of the *Business Rules Manifesto* states this in very clear terms "*Of, By, and For Business People, Not IT People*." As owners of business rules, only business people can create, modify, or state that a rule is no longer valid.
- **Declarative**—Business rules are declarative and not stated procedurally. The rule is declared; how the rule is enforced isn't part of the rule.
- **Separate from Processes**—Rules are not processes or procedures. They apply across processes and procedures and should not be contained within them. There should be one cohesive body of rules, enforced consistently across all business activities.
- **Precise**—A business rule must be open to only one interpretation. If the rule can be understood to mean more than one thing, you have to restate it.
- **Atomic**—A business rule contains a single complete thought, but not more than one. The business rule must be indivisible; if you try to break up a true business rule into parts, you'll lose information. However, just because the rules are atomic does not mean that the business rules do not build on each other.
- **Consistent**—A set of business rules must not contain conflicting rules.
- **Non-redundant**—A set of business rules must not contain rules stating the same information.

5. **Business Rules Types.** Authoring teams will also need to be educated in the various types of business rules categorized by function. The six business rules types included in Enterprise Portfolio™ are as follows:

- **Terms**—The first type of business rules to be discussed are terms. A *term* is a word, phrase, or acronym that has a specific meaning for an organization. More often than not, organizations find that they need to maintain a repository of terms that aren't part of everyday speech. Formalizing these terms is logical and will be profoundly helpful. As previously mentioned, business rules are meant to be "of, by, and for business people, not IT people." This means that industry specific jargon shouldn't have a place in business rules.

  As a business rule, terms are nouns or noun phrases with an agreed upon definition. Included in these should be core and unchanging concepts. The definition provided for a term describes an item of business interests or concepts. Specific business data is often incorporated into the business rules and these nouns have to be redefined in business friendly language.

  A formal vocabulary is applied to candidate business rules to ensure the terms and definitions are correctly expressed and do not conflict with other definitions. For example, the term bimonthly is often problematic. Some sources, including most dictionaries, define bimonthly as either meaning twice a month or every two months. In some cases, a customer or organization's name might even be considered a term. Providing defined terms helps prevent potentially disastrous misunderstandings.

The meaning of these concepts is given by definitions. When you look at a fact rule type, you should see a structured business vocabulary that includes both nouns (terms) and verbs (wordings). These nouns and verbs are used directly in expressing business rules. If business rules are instead expressed as an abstract knowledge statement, the rules become inscrutable because the meanings of the symbols or language are deeply nested. The extra effort to translate this must take place far in advance of executable applications.

Business rules without business vocabulary cannot facilitate decisions effectively, since no clear definition is given to the concepts handled by rules. As a result of this, rules may lead to unexpected or wrong decisions from the point of view of the stakeholders. As one can see, building a working business vocabulary is a necessity.

- **Facts**—To put it simply, *facts* are true assertions about a business; they can describe associations or relationships between business terms. They have been called invariants, immutable truths about data entities and their attributes. While many business rules often accredit specific facts, facts usually do not translate directly into functional requirements. Facts about data entities that are important to the system might appear in data models that the analyst or database designer creates.

  Moreover, facts accentuate the merger of multiple terms, so they characterize the relationships between terms. If further data becomes necessary, the relationships between terms can be qualified by other labels.

  Examples of facts are:

  - Every chemical container has a unique bar code identifier.
  - Every order has a shipping charge.
  - Each line item in an order is a specific combination of chemical, grade, container size, and number of containers.
  - Nonrefundable tickets are subject to a fee when the purchaser changes the itinerary.
  - Sales Tax is not computed on shipping charges.

  A good rule of thumb to follow is that facts often include a *be* verb in its present tense. This is discounting the word *am*, which is also a present tense be verb, but it is first-person singular and is rarely applicable here. The *be* verbs that need to be used in facts are: *is* and *are*. Specifically, *is* is used in third-person instances and *are* is used in second-person and first, second, and third person plural instances. One reason why these *be* verbs are used in facts is that they can be used in affirmative and negative sentences.

  Other words can also indicate a fact; the words *is* and *are* are just the most common. Another common word used in facts is *has* or *have*. Take this example: “every shipping container has a unique bar code identifier.” This data is clearly represented as a fact, despite missing the *be* verb.

If one were to literally define what a fact is, they would see that it refers to something that has either really occurred or has an actual, provable existence; facts are parcels of information that are presented as having an entirely objective reality. If an organization is unsure if one of their rules is a fact or not, an easy indicator is whether the stated fact is verifiable. If the rule is provable, it's a fact. Take these basic examples:

1. A customer is classified as commercial, government, or non-profit.
2. Commercial customers must have a valid tax identification.

At first glance, both examples presented appear to be facts. Unfortunately, only one of them matches the requisites necessary to being a fact. The first example does possess the word *is*, and the statement is affirmative and delivered without any ambiguity or subjectivity. The first example simply informs us what a customer is, and is a clear representation of a fact.

The second example, however, is not a fact. Firstly, it is missing *is*, *are*, or another indicative word. Secondly, the example does not provide a provable absolute. There might be instances where commercial customers do not yet have valid tax identification and need to resolve the issue. Thirdly, the keyword to notice in the second example is *must*, a word usually associated with an entirely different rule type: constraints.

- **Constraints**—A *constraint* is a rule statement that concerns dynamic aspects of an organization and defines allowable actions, specifically allowable changes to data. Constraints are conditions that have to be satisfied by data in order to store it into record. In one way or another, every organization constrains behavior, and this is closely related to constraints on what data may or may not be updated. Preventing records from being made is often the reason for a constraint; no record usually prevents actions from taking place, as well.

  Organizations usually have security and policies used to control access to their information systems. These policies often declare that passwords should or will be used. In addition, these policies present rules regarding frequency of required password changes and whether or not previous passwords may be reused, etc. These are constraints about application access that could be considered business rules. Tracing each such rule into the specific code that implements it makes it easier to update systems to comply with changes in the rules, such as changing the required frequency of password changes from 90 days to 30 days.

  Software projects often have myriad constraints. Project managers find it necessary to work within schedule, staff, and budget limitations. These project-level constraints belong in the software project management plan. Product level design and implementation constraints that restrict the choices available to the developer belong in the SRS or the design specification. Many business rules impose constraints on the way the business operates. Whenever these constraints are reflected in the software functional requirements, organizations should indicate the pertinent rule as the rationale for each such derived requirement.

Constraints are described non-procedurally, in terms of other atomic business rules. When a constraint describes a possibility, it imposes limitations and restrictions. Constraints restrict the actions that the system or its users may perform. Constraints can do this through their word choice. Some words and phrases that suggest someone is describing a constraint business rule are: must, must not, may, may not, and only. The words must is an easy indicator of a constraint since it means that something is obliged.

Examples of constraints are:

- A borrower who is less than 18 years old must have a parent or a legal guardian as cosigner on the loan.
- A library patron may place up to 10 items on hold.
- A user may request a chemical on the level 1 hazard list only if he has had hazardous-chemical training within the past 12 months.
- All software applications must comply with government regulations for usage by visually impaired persons.
- Correspondence may not display more than four digits of the policy holder's social security number.
- Commercial airline flight crews must receive at least 8 hours of continuous rest in every 24-hour period.
- The attending physician must certify that the principal and secondary diagnoses in the medical record of each patient discharged is accurate.

If we look closer at the last example presented, we can see the business rule *attending physician* (which is a term) is the anchor object of this constraint. The business rule that expresses the possibility that an attending physician certifies principal and secondary diagnoses of discharged patients (a fact) is the correspondent object of this constraint.

Furthermore, constraints can be classified into four separate classes. These classes are divided as types, relations, and data.

- **Type Constraint**—Type constraints are typically just definitions of the set values that compose the type in question. This constraint type should not ever make mention of any variables that might exist.
- **Relation Constraint**—Relation constraints are constraints simply defined by its relations. These often illustrate a one to many relationship. An example of this is: an order must have at least one line item, but may have up to 99,999.
- **Data Constraint**—These constraints constrain the values a given set of data is permitted to assume.

It is important to note that constraints are also often referred to as action assertions.

- **Action Enablers**—*Action enablers* are the rules that trigger activities under specific conditions. Action enablers can be performed in a manual process. Alternatively, the rule might lead to specifying some software functionality that makes an application exhibit the correct behavior when the specified conditions are true. The conditions that lead to the action could be complex combinations of true and false values for multiple individual conditions. A decision table can provide a concise way to document action-enabling business rules that involve extensive logic. A statement in the form "If <some condition is true or some event takes place>, then <something happens>" is a clue that someone is describing an action enabler. Following are some examples of action-enabling business rules.
  - → If the chemical stockroom has containers of a requested chemical in stock, then offer existing containers to the requester.
  - → If the expiration date for a chemical container has been reached, then notify the person who currently possesses that container.
  - → On the last day of a calendar quarter, generate the mandated OSHA and EPA reports on chemical handling and disposal for that quarter.
  - → If the customer ordered a book by an author who has written multiple books, then offer the author's other books to the customer before accepting the order.

  It might be obvious to some that action enablers are expressed as if-then rules. This refers to if as a condition and then as an action. Unfortunately, other rules are often passed off as action enablers, leading to confusion. A reason for this is that it is possible for the other categories, like calculations and constraints, to be expressed as as an *if-then* rule. There are two main ways to tell the difference between a true action enabler and another rule trying to pass itself off as an action enabler.

  Primary Difference: There is a type of action being performed in the action part of the rule. With computations and inferences, the action part sets the value of an attribute. With constraints, the action part typically records the failure of a certain condition. By contrast, with action enablers, the action part launches a domain meaningful process, as opposed to a value setter.

  Secondary Difference: The impact of firing an action enabler is very different from the firing of a constraint. Usually, as action enabler launches secondary business processes as the main process continues. By contrast, firing a constraint violation rule puts an end to a business process.

- **Calculations**—The next class of business rules are *calculations*. These rules describe business rules that involve calculating values following explicitly specified arithmetic operations. *Calculations* are exact formulae and algorithms for how to calculate computed terms. Often, calculations are used to follow external rules, like income tax withholding formulas. It is necessary for calculations to be able to be traced to an activity in the workflow and, eventually, to an operation performed by a business worker or a business entity.

  Calculation rules can be evaluated to actually produce results. The calculation may even be inadequate or mis-specified, but that is more representative of poorly developed guidance, not any violation. This would make calculations more structural, rather than operative.

  Calculations can be written in a narrative form. The following are examples of this:

  - → Requests must not exceed a client's available budget.
  - → A customer's annual order volume must be computed as total sales closed during the company's fiscal year.
  - → The total price of an order item is always computed as the product unit price times its quantity.
  - → Domestic ground shipping charges for orders that weigh more than two pounds is $4.75 plus 12 cents per ounce or fraction thereof.
  - → The amount charged to a customer for an order item must be equal to the total price of that order item.
  - → Unit prices are reduced by 10 percent for orders of 6 to 10 units, by 20 percent for orders of 11 to 20 units, and by 35 percent for orders of more than 20 units.

  After viewing the examples presented above, it is obvious that some calculations can become awkward when written in a narrative form. This awkwardness will usually happen if the calculation presented has a large amount of numbers and mathematical symbols. Using a narrative form for calculations can result in wordy or overly complicated sentences, but there are other ways to document these rules. Often, organizations present these rules as mathematical expressions. In addition, these organizations also commonly use tables to display calculations in a clear way and allow for easier maintenance.

Watch out for boundary value overlaps when writing sets of business rules that define ranges. If an organization defines ranges like 1-5, 5-10, and 10-15, there will undoubtedly be future issues concerning the ranges where numbers 5 or 10 should fit.

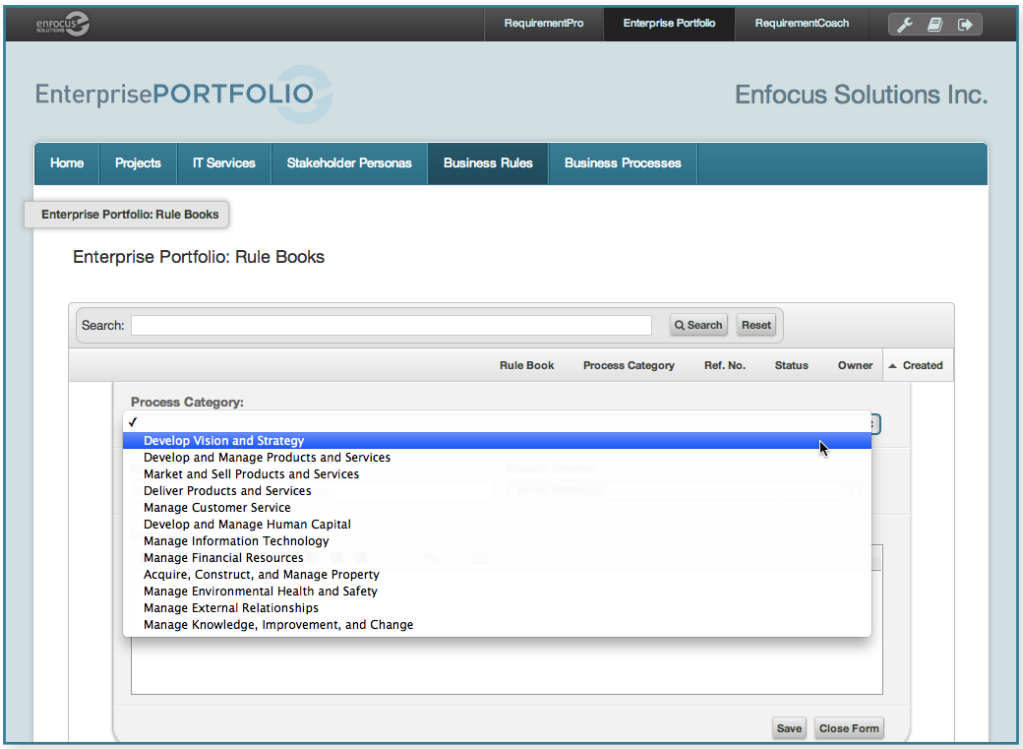
- **Decision Tables**—Decision table rules require a decision table for implementation and are essentially business rules for making decisions. Decision tables are valuable for displaying cause and effect relationships, examining multiple paths, and illustrating a set of conditions resulting in actions. Decision tables are useful for writing executable code, as well.

6. **Evaluate Business Rules.** The business analyst will next evaluate business rules written by the authoring team and have other stakeholders evaluate the business rules to ensure that they are well rounded, understandable, and soundly written. This can be done individually as the rules are written, or review sessions can be held with each authoring team to present and review the rules together.

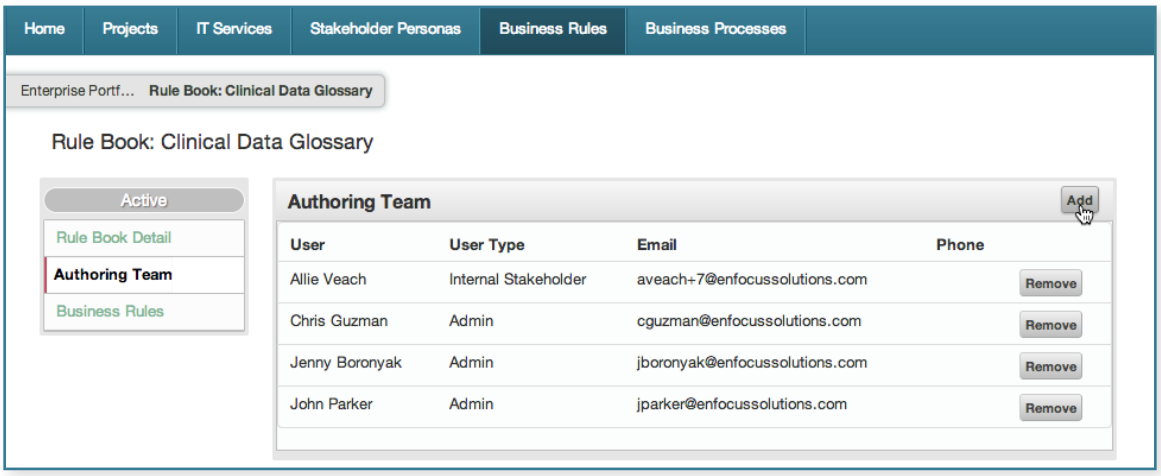
7. **Input Rules Into Enterprise Portfolio™.** Enterprise Portfolio™ will maintain a record of all rules, whether or not they are automated, so that when business rules change, organizations no longer have to update each project individually. Essentially, organizations are able to update repositories once and then trace the effects of changes to each project.

Adding gathered business rules to RequirementPro™ (Enterprise Portfolio?) is a simple process. First, go to the Enterprise Portfolio™ by selecting its tab at the top of the page. Once in the Enterprise Portfolio, select the *Business Rules* tab. In the Business Rules section, you can press the *New* button to create a new rule book, or you can simply select a rule book to make edits or add rules.

When creating a new rule book, you can choose its name, process category, status, record owner, and description, as shown below:

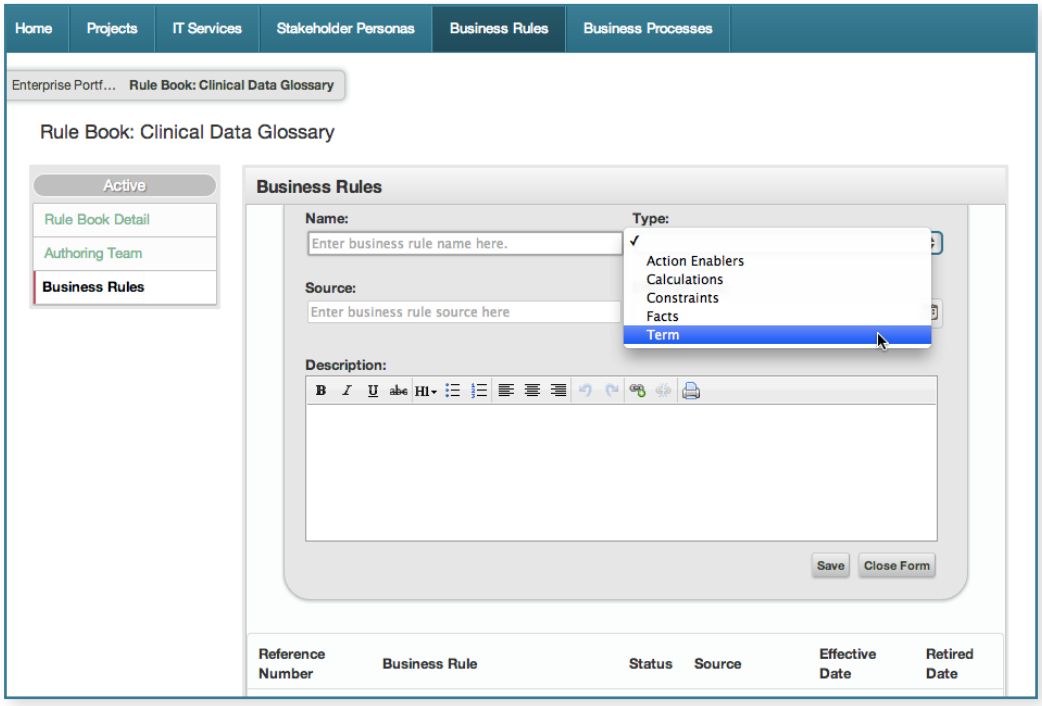


After a rule book is created, the business analyst can add attachments and comments that are necessary and helpful in the Rule Book Details. The next logical step is to document the authoring team associated with the glossary. This can be done by selecting the *Authoring Team* tab located underneath *Rule Book Details* on the left hand of the screen as shown:



After adding your authoring team, the authors should now populate your rule books with your gathered rules. To add business rules, select the business rules tab, also located on the left hand of the screen. The authoring team can create and enter new rules by selecting the *New* button, and editing a previously included business rule is as easy as selecting the rule you would like to edit. When adding business rules, the author can input the following fields:

- Name
- Type
- Source
- Effective Date
- Description



Collaboration

Gathering business rules is a collaborative effort; no one person can or should attempt this task alone. Considering that business rules include corporate policies, government regulations, industry standards (such as accounting practices), and computational algorithms, you can see why BAs shouldn't be the only ones writing the rules.

Software teams will need more collaboration with BAs and stakeholders to guide them through uncovering, analyzing, capturing business rules. Without this assistance, developers might simply make whatever assumptions are needed to write the code. This could possibly include building their assumptions into the software with little regard to the impacts on business. Inadequately capturing business rules creates rework and other inefficiencies.

Collaborating while writing and validating business rules is instrumental to ensuring that everyone understands the rules. Having multiple people working on rules helps prevent over complexity. If it takes multiple doctoral degrees to understand a business rule, then it has been written poorly and there is always the possibility of miscommunications and issues.

Multiple points of view are also helpful in keeping business rules business oriented. Collaboration should help in developing universal terms. This is necessary as business rules should be meaningful, understood, and confirmed across the business domain. Additionally, collaboration will reduce conflicting or self-contradictory rules.

It is crucial that collaboration is encouraged for maintaining rule books. Many organizations neglect documenting their business rules or forget that they are invaluable assets. When these organizations try to document their rules, their efforts suffer from insufficient collaboration and documentation, and their rules are ineffective.

It is also extremely important that everyone is aware of the rule book's existence. If people remain uninformed, there are dangerous side-effects. Additionally, the more people who collaborate and are well-informed of the business rules, the less worrisome it is when people part ways with your organization, i.e., if only one person knows about a business rule, the rule itself can be lost after the person is gone.

Perhaps most importantly, remember that business rules are owned by the business and should always be created and maintained by business people seeking active collaboration.

Visualization and Elaboration

Decision tables and decision trees are two different techniques for representing what a system will do in the face of complex logic. However, it is not necessary to create both a decision table and a decision tree to show the same information; either one will suffice. While decision trees and decision tables represent the same information, both visualization methods have their advantages and disadvantages. The biggest advantage to decision tables is that they are detailed and precise enough to ensure completeness and accuracy. However, decision trees are easily understood and present a more graphic view of logical processes.

Decision table and trees are used as visualizations when figuring the implications and process of a Decision Table type business rule.

Decision Tables

When used by Business Analysts, decision tables allow for the creation and use of easily understandable business rules. Decision tables are used when there are certain conditions that must be evaluated and specific actions occur when those conditions are met. According to the book, Business Analysis Techniques, "A decision table shows a set of conditions that may be combined in different ways in order to determine the required courses of action. Decision tables provide a clear and unambiguous means of documenting conditions and the resultant actions to be taken." In essence, decision tables are a compact way of representing a complete set of rules.

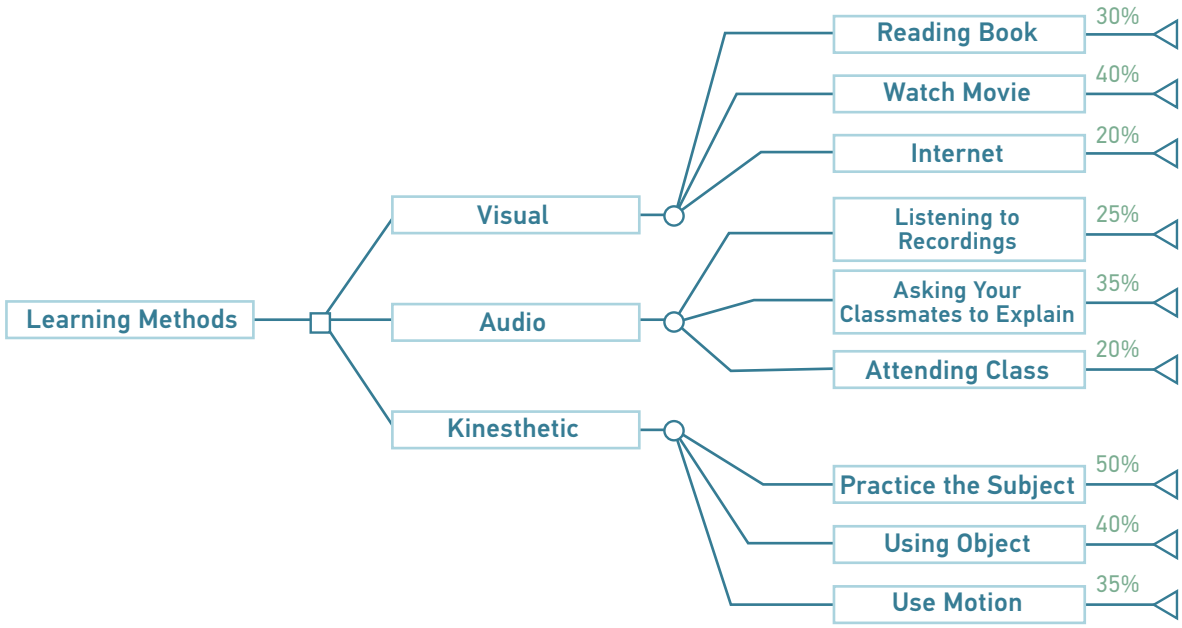
The completed decision table below is from the book, *Business Analysis Techniques*:

Condition 1	Travelling after 10 am?	Y	Y	N	N
Condition 2	Aged 60 or over?	Y	N	Y	N
Action 1	Off-peak price less 20% discount	✓			
Action 2	Off-peak price		✓		
Action 3	Full price less 20% discount			✓	
Action 4	Full price				✓

Decision Trees

Decision Trees provide a tree-like graph to quantify and visually display decisions and their possible outcomes. They provide an extremely efficient structure to lay out options and explore the possible outcomes of choosing those options. Decision Trees also help you to form a well-adjusted visual of the risks and rewards associated with each possible course of action because competing alternatives are calculated. Through these calculations, decision trees indicate which decision has the highest utility, or is the most preferred, to the decision maker. Adding numbers in the form of probabilities, expected payoffs, and expected costs can further customize decision trees and add to more in depth decision tree analysis.

Below is a decision tree example exploring the learning methods available to schools.



Task Completion Checklist

The following task completion checklist may be used to ensure that the business analyst has accomplished all goals associated with this task.

	✓
Do your business rules help reduce costs?	
Do your business rules help further agility?	
Do your business rules help protect revenue?	
Do your business rules help minimize risk exposure?	
Do your business rules help market share maintenance?	
Do your business rules help refine decision-making?	
Do your business rules help in business processes?	
Do your business rules automate policies and decision?	
Do your IT resources become freed up when business users are able to edit rules by themselves?	
Do your business rules help constrain business activities?	
Do your business rules help create a logical framework for business operations?	
Are your business rules easily changed, as needed?	
Are your business rules easily understood?	
Are your business rules as comprehensive as possible, leaving little room for guesswork?	

4.5 Chapter Five
Document Terminology

Introduction

There is bound to be cause for conflict when people involved in a project have multiple interpretations for a single term. A way to avoid this is to ensure that everyone involved in the project has an understanding of potential terms. Terms are words or expressions that have precise meaning or are peculiar to a specific field. Terms can be different for developers and non-developers, and using standard terms can be challenging due to misunderstandings and consistency issues. It is a best practice for organizations to provide a uniform glossary of terms intended for common use. The goal of a glossary is to provide an organization with a common vocabulary that all stakeholders can agree upon. Glossaries help define the meaning of business terms relevant to not just one system or project, but to an entire organization.

Creating a strong glossary early on in an organization's lifetime or at the beginning of a project including organization wide terms, business terms, acronyms, and domain specific terms has immense benefits. Wasted time, conflict, and confusion can easily be avoided with an accurate, concise, easily understandable, and easily accessible glossary in place. Having a complete and editable glossary is important because terms included will serve as a base for requirements and business rules. Without an agreed upon understanding of a term, a requirement could be misunderstood. This misunderstood requirement would then decrease the project's value because time is wasted going through development and validation before it's realized that the requirement was unintended and delivers unintended benefits. Since a glossary is a collection of terms and their definitions, it should also contain abbreviations and acronyms, synonyms, homonyms, technical terms, and non-technical terms.

Core Concepts

Begin the Glossary Early

It is a best practice to begin glossaries as early on in a project or organization's lifetime to avoid later miscommunications and difficulties. People can find definitions for any term they need by searching the web. Unfortunately, these searches require time and effort, and the quality of results is unpredictable because definitions are pooled from different sources and contain multiple inconsistencies.

Glossaries Need to Be Maintained

Glossaries are a live document that should be reviewed and updated on a regular basis. Failing to do so can result in outdated or undocumented terms going forward past the initial setup of the glossary. Many times, terms become obsolete or new terms and acronyms are introduced into the business, especially when a new solution is being implemented – be sure to have a glossary maintenance process in place to avoid an incomplete or inaccurate glossary.

Glossaries Need to Be Communicated

A glossary is only effective if people know it is available to use, and can be trusted as a single source of reference. Part of building that trust is to make the business a large part of building it, communicating the value, and introducing the glossary or glossaries into the business in a way that sets it in place as the go-to reference for terminology in the company. Keep your glossaries accessible and reference them often during other activities in the project (meetings, training, etc.) to display the glossaries' value.

Process

1. **Determine the Glossary Anatomy.** Decide on how the glossary is going to be set up. A glossary may be made up of one large document, or a set of smaller documents categorized by function, business area, department, etc. How a company sets up their glossary will depend on how the organization functions, the types of projects being undertaken, or whatever will be easiest for stakeholders to access and navigate. To determine the best way to set up your glossary, talk to your business about how they will want to use the glossary; how will they naturally search for terms?

2. **Ensure It Includes the Entire Organization.** Whenever setting up a glossary, don't do it with just the current project in mind. Think about what's best for the organization as a whole so that the glossary may be made for future projects. Be sure to consider the entire organization when deciding on the glossary's set up. This probably means talking to stakeholders outside of the project you are currently working on when initially setting up the project. As a starting point, use an organization chart to determine what business areas exist, and who the key stakeholders from each will be to include in the process. You want to make sure that how you initially setup your glossaries will be of value to everyone.

3. **Provide Sources for Terms.** Determine what the best sources will be for defining and identifying your terms. A good place to start when identifying sources for terms is documents. The kinds of documents that terms may come out of are the documents dealt with in *Task 4.2 Analyzing Documents*. Another source will be business reports in particular. Business reports are especially helpful in identifying terms because they are often short and full of useful terminology and acronyms. Acronyms are words, such as SCUBA, formed from the first letter or letters of each of the major parts of a compound term. Most glossaries also group initialisms, such as F.B.I., with their acronyms. Stakeholders are valuable term sources, as well. A glossary is supposed to be a trustworthy place that people can go to for accurate and correct information, so spend time determining which sources will provide the most informed and accurate information.

4. **Follow Our Glossary Guidelines.** When creating a glossary from scratch, there are a few best practices and guidelines recommended by the analysts at Enfocus Solutions Inc. for a successful glossary:

   → **Centrally Managed.** It is recommended that glossaries be centrally managed. There should only be one glossory or a set of glossaries to manage terms. If there are multiple valid glossaries, then people will lose trust in the definitions; How will they know which definition is the correct one?

   → **Maintained with Consistency.** It is recommended that glossaries are maintained for consistency. If more than one person is maintaining a glossary, ensure that everyone is on the same page and follows a glossary maintenance process.

   → **Long Term Maintenance.** It is recommended that a process is set in place for the addition and editing of existing terms. There should also be a process for deleting old or irrelevant terms.

   → **Accessible.** Glossaries are not of value to people if they are not accessible.

   → **Contains Sources.** Including the sources in your glossary allows people to go back to the term's source if any questions arise. Sources may be revisited for validation of a term or problem solving.

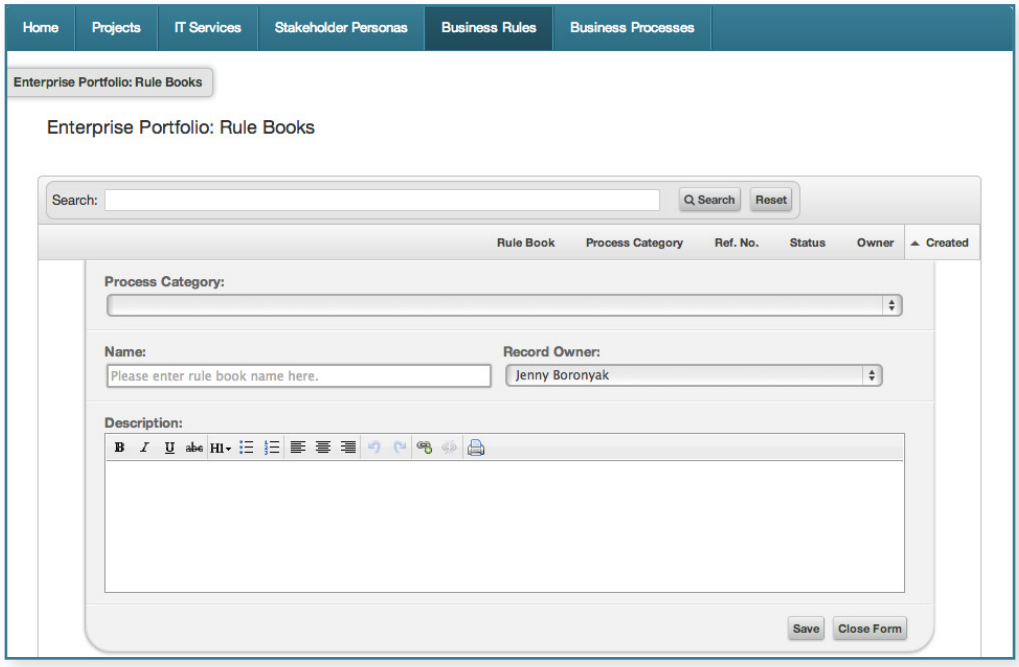
   → **Agreed Upon.** Make sure the stakeholders and the business understand and agree upon terms and their definition.

5. **Write the Terms.** The next step is to write the terms to be included in your glossary, which will be done on an iterative basis as terms are discovered. Below are rules and guidelines to keep in mind while writing:

   - More than one definition may be provided for a single term if necessary. This is helpful when terms are defined differently depending on perspective. Also, terms that have both a general meaning and a specific meaning in a requirements context should have both meanings defined and documented.
   - Provide enough detail and clarity. Terms containing details or ideas helping to explain the term to multiple audiences may be necessary.
   - Include synonyms and homonyms where you think are needed to prevent confusion. Synonyms are two or more terms that have the same or similar meanings, and homonyms are two or more terms different in meaning, but spelled and pronounced alike.
   - Be consistent. By defining terms, you considerably increase the understandability of requirements. You can avoid misunderstandings and different interpretations of terms that might lead to conflicts if terms are consistently defined and structured.

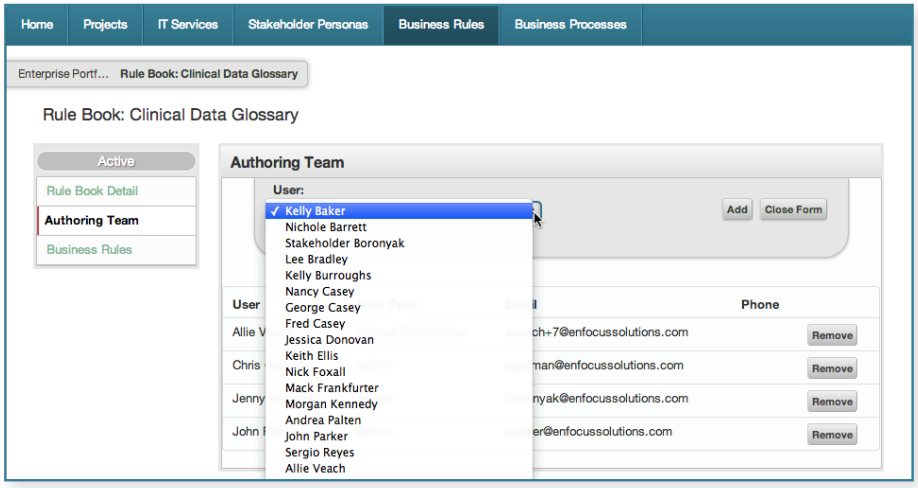
6. **Add Glossaries to RequirementPro™.** Adding glossaries to RequirementPro™ is a simple process. Glossaries are contained in Rule Books, and terms are documented as individual business rules. To create a glossary as a rule book, go to Enterprise Portfolio by selecting its tab at the top of the page and then select the *Business Rules* tab. In the Business Rules section, you can press the *New* tab to create a new Glossary, or you can edit a previously created glossary.

When creating a new Glossary, you should input its process category, name, record owner, and description.



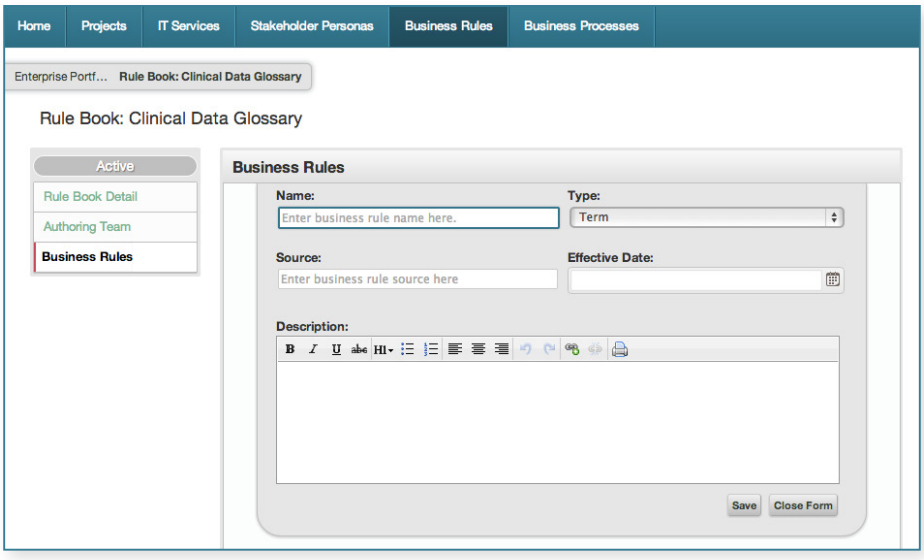
After creating a glossary, attachments and comments may be added in the Rule Book Details section if necessary.

7. **Document Authoring Team in RequirementPro™.** After this, the next logical step is to document the authoring team associated with the glossary. This can be done by selecting the *Authoring Team* tab as shown below:



8. **Add Terms to RequirementPro™.** After adding your authoring team, the glossary may be populated with terms and acronyms. Terms may be added by selecting the Business Rules tab. Select an existing term to edit it, or create new terms by selecting the **New** button. Terms may also be imported using the business rules Import button.

   When adding terms, input the term name, select the type "Term" from the type drop down menu, input the term source, and then enter the term's definition in the description field. The effective date field may be left blank.



9. **Follow a Quality Control and Approval Process.** After terms have been added to a glossary or glossaries in RequirementPro™, collaboration is necessary to review them for consistency and formatting, among other things. Stakeholders should also review terms for clarity, conciseness, and understandability. Involving multiple stakeholders in a glossary quality control and approval check will make for successful future glossary use.

10. **Publish the Glossary/Glossaries.** Once a glossary is completed, a message should be sent out advertising that the terms in the glossary are accurate and that the glossary should be referenced often. Making an announcement about the usability and benefits of using a glossary will increase a glossary's use, and, therefore, its organization wide benefits.

11. **Develop the Glossary Maintenance Plan.** A process for adding and editing terms in glossaries should be set in place to ensure that the glossary/glossaries remain up to date, include relevant terms, and contain continuously accurate definitions. Who can add and edit terms, and under what circumstances a term may be added should be established. Consider what will happen when new terms are discovered – who will search to see if they currently exist? If they do exist, who will review if it is accurate or needs to be updated? Or if it doesn't exist, who will be responsible for adding it and reviewing it? All of these are questions you can ask in order to determine what the best maintenance process is for your glossaries. The key is to actually have a process; a glossary which is developed and then left inaccessible or is left by the way side cannot be valuable.

Collaboration

Often, a cause for conflicts occurs when people involved in the development process have different interpretations of terms. To avoid these conflicts, everyone involved in the development process should share an understanding of terminology used.

Defining terms and creating glossaries should always be a collaborative process. As we mentioned earlier, terms and glossaries need to be agreed upon by the individuals who use them. To reiterate, since only stakeholders can reliably validate the definitions for their context, the stakeholders should validate each definition. Also, each individual definition in the glossary should be approved. This approval ensures that the term is correct and should be obligatory. Remind stakeholders that they should provide extra comments to a term if they think they can help define it.

A glossary can serve as a focal point for collaboration, especially if each stakeholder contributes terms, definitions, or comments on submitted definitions. When stakeholders are responsible for creating terms and definitions, they are much more likely to remember the term and the correct definition. Even just engaging stakeholders with learning, debating, and refining a glossary can go a long way toward helping them understand new terms.

Task Completion Checklist

The following questions may be asked to determine if you have successfully implemented the process described in this task, document terminology.

	✓
Does the glossary reduce ambiguity?	
Are the terms included concise, or could they be used or interpreted in multiple ways?	
Does the glossary help to create mutual understandings amongst stakeholders?	
Does the glossary speed up communications?	
Are abstract terms clearly defined?	
Are abstract terms used consistently?	
Has each term been reviewed by stakeholders?	
Has the glossary been published and advertised within the company?	
Is the glossary easily accessible to users and logically structured?	
Is there a plan in place to maintain the glossary long-term?	
Does the organization know how to access the glossary? Search for a term?	
Is the organization aware of how the glossary will be maintained? What to do when they want to add a new term?	
Does the organization trust the glossaries as a single source of terminology reference?	

4.6 Chapter Six
Gather Assumptions

Introduction

Assumptions are important to the success of a project, because without them the business analyst and project contributors would be unable to manage a project's unknowns. An assumption is a theory pertaining to the project that has been declared to be true, as if it were true. If you do not successfully gather and manage your assumptions, they will eventually *manage you* by causing problems, like cost overruns or other unexpected and unplanned behaviors. Note that assumptions may add risk to a project, since it is possible that they will turn out to be false. A good list of assumptions, however, is a framework to build a project upon that loosely defines scope and should be started on early in a project's timeline.

Gathering, evaluating, and describing what an assumption is and how to make quality assumptions is important for every project. Sometimes, aspects of a project have to be assumed before planning is even started. For example, the business analyst might assume that there will enough money to finish the project. Assumptions are not fixed; however, they are fluid and could possibly change as you move through the project. They can influence any part of your project life cycle and solution implementation, so it is important to document and analyze them. There are four types of assumptions gathered and input into RequirementPro™, but this chapter will focus on gathering and documenting solution assumptions. After writing, categorizing, and documenting solution assumptions, they will be used to identify potential risks that can affect a project's implementation and delivery; they will also be used to identify assumptions that have a negative effect on end-user expectations.

Core Concepts

Assumption Types and Categories

Assumptions are collected and managed most easily by being grouped into assumption types and categories. The analysts at Enfocus Solutions Inc. have broken down assumptions into four different types: business analysis assumptions, business case assumptions, project assumptions, and solution assumptions. Each of these assumption types are then further grouped into categories and aid in organization. Listed below are descriptions of the four types of assumptions and what information is contained in each assumption category.

1. **Business Analysis Assumptions.** Business analysis assumptions support the recommended business analysis technique or approach being employed throughout a project. Many business analysis assumptions are elicitation method assumptions.

   There are eight, business analysis assumption categories collected throughout the tasks in *Task Group 1.0 Business Analysis Value Management*.

- **BA Roles and Responsibilities Assumptions** will contain information about the business analyst's functions.
- **Conceptualization Approach Assumptions** will contain information about activities that need to be performed to determine what the solution will look like.
- **Elicitation Approach Assumptions** will contain information about stakeholder elicitation. For example, do you assume that needs, scenarios, or both will be gathered for stakeholder elicitation?
- **Visualization Approach Assumptions** will contain information about which types of visualizations will be necessary for organization wide comprehension, when necessary.
- **Bundling Approach Assumptions** will contain information about the approach that will be taken to begin bundling requirements.
- **Requirements Management Approach Assumptions** will contain information about how requirements are to be managed after being bundled. For example, how will you manage changes to requirements?
- **Socialization Approach Assumptions** will contain information about how stakeholders are to be engaged for collaboration. Information about how the business analyst is going to understand stakeholders or document stakeholders' experiences are in this category, as well.
- **Planning and Managing BA Activities Assumptions** will include information about the determination of a project schedule of events, and information about deliverables.

2. **Business Case Assumptions.** Business case assumptions are developed in *Task 3.6 Define Business Case*. These assumptions are made by a business analyst while developing the business case to support financial projections. Funding and risk management assumptions would fall under this assumption type, as well. Typically, any assumptions including financial information are business case assumptions.

3. **Project Assumptions.** Project assumptions are assumptions made to determine how a project will be run. These assumptions will be made by the business analyst and then given to a project manager, as they are focused around the project team and environment.

There are four categories of project assumptions.

- **Project Team Assumptions** will contain information about team members, such as their skill individual skill level and business domain.
- **Physical Space and Resource Assumptions** will contain information about office space, computers and network, telephones, and other equipment that may be needed by a project team.
- **Time Frame and Deadline Assumptions** will contain information about project deadlines and time frames.
- **Communications and Collaboration Assumptions** will contain information about things like regularly scheduled meetings or reporting processes.

4. **Solution Assumptions.** Solution assumptions are the assumptions we will be developing in this task, *Task 4.6 Gather Assumptions*. This assumption type includes assumptions made to execute a solution and focuses mostly on users.

   Depending on the solution, there are seven categories of solution assumptions.

   - **User Assumptions** will contain information about the amount of people using a solution, whether training will be needed for these users, or where the users are located.
   - **Compliance and Regulatory Assumptions** will contain information about required standards and regulations, legal standards, and company policies.
   - **Interfaces and Dependencies Assumptions** will contain information about system interfaces and dependencies with other projects.
   - **Technology Assumptions** will contain information about network availability, operating systems, and system requirements needed for the solution.
   - **Business Process Change Assumptions** will contain information about the amount of change needed to implement a solution into the current process.
   - **Solution Acquisition Strategy Assumptions** will contain information about whether a solution will be purchased or built.
   - **Testing Assumptions** will contain information about what types of testing, and how much testing will be necessary.

Business Analysts and Project Managers Use Assumptions

The business analyst uses business case assumptions mainly when creating a business case. For more information on business cases, see *Task 3.6 Define Business Case*. At the beginning of a project, while developing the rationalization for the project, there will be a ton of unknowns, and assumptions help to construct the initial business case for the project. Solution assumptions are used in much the same way. The business analyst will create solution assumptions for the business end of management, and then give them to project managers. Project managers, on the other hand, will utilize assumptions primarily with the project charter and to help build the schedule. Even though assumptions are used by different positions to accomplish different goals, assumptions should be created using the process described in this task.

Assumptions are Established Early and Evolve with the Project

Assumptions are to be established early on in a project and will be changed and modified as the project progresses. It's important that assumptions are created early on in a project, because they are used to establish the project environment and scope, and provide a foundation for planning and estimating. However, assumptions are only accurate for short periods of time because as a project is developed and details are established, the assumptions will inevitably change, too. As project changes occur, it is hard to determine the exact impact of the change on the project or product, so assumptions are reformulated or deleted periodically.

Assumptions Should Be Verified

Given assumptions are unknowns only presumed to be true, they need to be reviewed regularly and verified as the project progresses in the event they are not actually true. Assumptions influence how projects are approached, and also the requirements that we define and build. Not verifying assumptions to actually be true can put at risk the project work and requirements that are influenced by them. For example, an initial assumption may be that the new solution will only introduce a moderate amount of change into the existing business processes; however, it may be discovered later on that actually it will introduce a more significant amount of change than originally thought. The impact of not reviewing and verifying that assumption is that the business cannot be properly prepared for the change, and that the solution will not actually function as the business originally expected – both of which put solution adoption at risk.

Process

Solution assumptions are made for projects simply because, at the time of planning it is impossible to know everything. This could be because the facts are not yet fully known or not yet decided. Assumptions could also be necessary because the project is dependent on the outcomes from other projects, for example, the delivery of a consultation document, or the construction of a new policy.

Solution assumptions are especially valuable because they provide a framework for you to build a project upon, and allow for the building of a project or product in the face of uncertainties and unknowns. The following process for gathering assumptions can be adapted to gather any of the four types of assumptions, but is specific to gathering solution assumptions.

1. **Write Assumptions.** Project managers, project sponsors, business analysts, and project members should all collaborate to write a solid set of solution assumptions. Assumptions are gathered from stakeholder experience, prior knowledge, and the information currently available, so it is important to have a plan for gathering them.

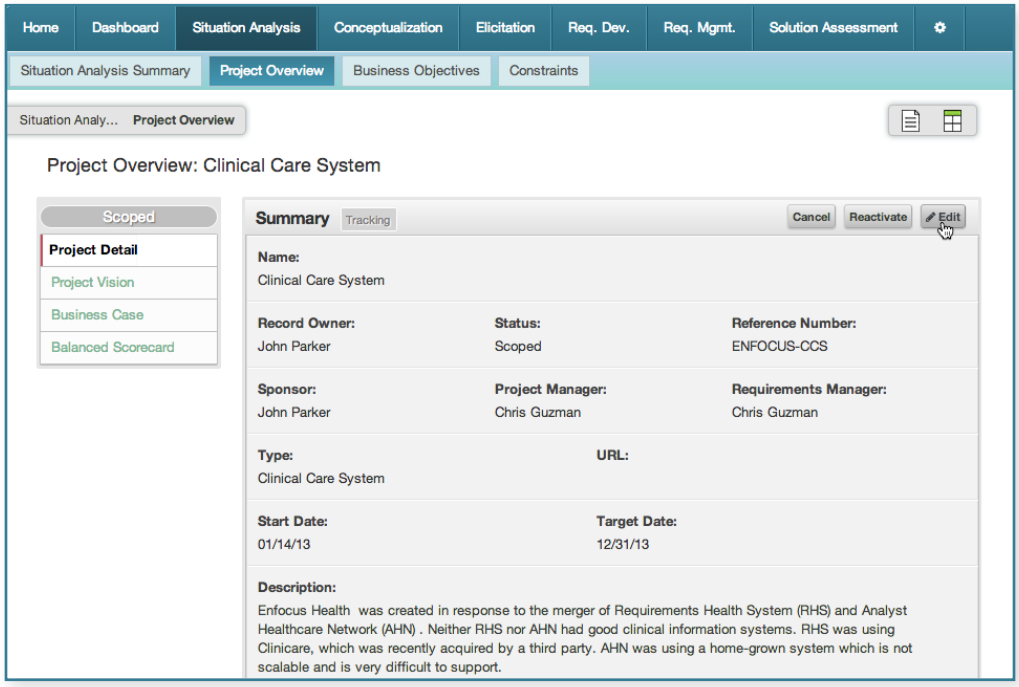
   A project meeting should be held to discuss and write assumptions. Brainstorming with mind maps is useful for fostering discussion and a healthy flow of ideas, and interviews should be conducted with project managers, project sponsors, and key project members to gather assumptions. When writing assumptions, keep in mind that a properly written assumption will be a specific statement that is concise and definitive. The following list contains a few more guidelines to follow when writing solid solution assumptions:

   - Solution assumptions contain assumptions necessary to execute a solution and are focused mostly on the users.
   - If there is no risk, an assumption becomes fact and is no longer an assumption.
   - Requirements may be discovered through solution assumptions; its important to check assumptions to see if any should be moved to the requirements section.
   - An assumption including a lot of data elements is often a requirement.
   - Remember that business rules are not assumptions.

2. **Categorize Assumptions.** Assumptions are divided into assumption types and categories for organization and traceability purposes. After gathering solution assumptions together and writing them down, you will find that managing assumptions will be easier if they are categorized and grouped together. Reviewing assumption categories contained in an assumption type can also help to reveal assumptions that might have been initially missed. There are seven categories of solution assumptions.

   - **User Assumptions.** User Assumptions will contain information about users. For example, they could estimate the amount of people using a solution, whether training will be needed for these users, or where the users are located and where training will be necessary.
   - **Compliance and Regulatory Assumptions.** Compliance and Regulatory Assumptions will contain information about required standards and regulations, legal standards, and company policies. For example, they could project that the company will continue to accept personal checks.
   - **Interfaces and Dependencies Assumptions.** Interfaces and dependencies assumptions will contain information about system interfaces and dependencies with other projects. For example, the student information system at a university will access the professor grade book system.
   - **Technology Assumptions.** Technology solution assumptions will contain information about the estimated network availability, operating systems to be used, and system requirements needed for the solution to be implemented.
   - **Amount of Business Process Change Assumptions.** These assumptions will contain information about the amount of change needed to implement a solution into the current business process.
   - **Solution Acquisition Strategy Assumptions.** Solution Acquisition Strategy Assumptions will contain information about whether a solution will be purchased or built. For example, the solution will be purchased by January 1, 2050.
   - **Testing Assumptions.** Testing assumptions will contain information about what types of testing, and how much testing will be necessary for a successful solution implementation. For example, the assumption that business stakeholders will test user friendliness.

3. **Add Solution Assumptions to RequirementPro™.** Solution Assumptions will be added to projects in the Description field on the project detail page. Navigate to the project detail page by clicking the Situation Analysis tab and then clicking the Project Overview tab as shown below. Enter assumptions into the Description field by clicking the Edit button. Include each assumption's category and type in the Description field, as well.



4. **Review and Validate Assumptions.** Once assumptions have been gathered, written, and input into RequirementPro™, assumptions should be reviewed by stakeholders in a massive review process and on an on-going basis. The purpose of the review will be to consider each assumption's quality, accuracy, and effectiveness while identifying assumption omissions. Because assumptions are fluid and could be proven wrong at any time, it is important to validate assumptions as the project progresses. Leaving an assumption unverified and without review increases the chances that the assumption is not true and will put the project at risk.

5. **Assumption and Risk Analysis.** The business analyst may perform Assumption and Risk Analysis at this time to determine additional project risk and ensure that a risk hasn't been recorded as an assumption. Figuring out the risk associated with an assumption and making assumptions into risks is as easy as placing assumptions into an If/Then statement like the one below:

"IF this assumption proves to be false, THEN the effect on the project would be …"

In the above format, the IF side of the statement explores the likelihood of the assumption being unsafe, and the THEN side of the statement explores what the outcome would be. The IF statement will reflect an assumption's probability, whereas the THEN statement will reflect a failed assumption's impact. In other words,

"IF this assumption is proven incorrect, THEN what will likely consequences for the project be?"

After making assumptions into IF/THEN statements, these statements can be analyzed to determine which assumptions are most likely to fail or be false, and the business analyst can act accordingly. Faulty or false assumptions could impact multiple project objectives and should be considered risks.

Iterative Review. On an iterative basis, assumptions need to reviewed and tested, especially assumptions key to the project's scope, timing, or costs. Assumptions may be deleted from RequirementPro™ as work is completed and assumptions are proven to be facts or failures. To keep a history of assumptions that have been verified and unverified, an assumptions status may be set to 'verified' for tracking purposes.

Collaboration

Business analysts, project managers, project sponsors, and stakeholders will find that it is necessary to collaborate throughout any assumption gathering process. The business analyst must work closely with all stakeholders to clarify and validate assumptions, even after assumptions have been gathered and written using stakeholder input, and iterative collaboration is necessary for assumption reviews and tests.

Project managers are responsible for communication and collaboration within a project, so the business analyst should closely collaborate with project managers for maximal internal communication and collaboration with necessary external parties. Collaboration with a wide variety of stakeholders helps to ensure that assumptions don't become facts. The business analyst should share the results of assumption verifications. If not, only the BA and project manager will know that an assumption has been proven false, and everyone else involved with the project will go on thinking an assumption is true. It is up to the business analyst to update the status of assumptions or get rid of them, and to communicate that process, as well.

Task Completion Checklist

Below is a checklist for the business analyst to determine if this *Task 4.6 Gather Assumptions* has been completed.

	✓
Are the assumptions in RequirementPro™ written according to the following guidelines?	
• Specific?	
• Statements?	
• Concise?	
• Definitive?	
Was collaboration used in every process step to gather solution assumptions?	
Are your assumptions categorized?	
Were all of the assumptions reviewed and validated?	
Did an Assumption and Risk Analysis take place?	