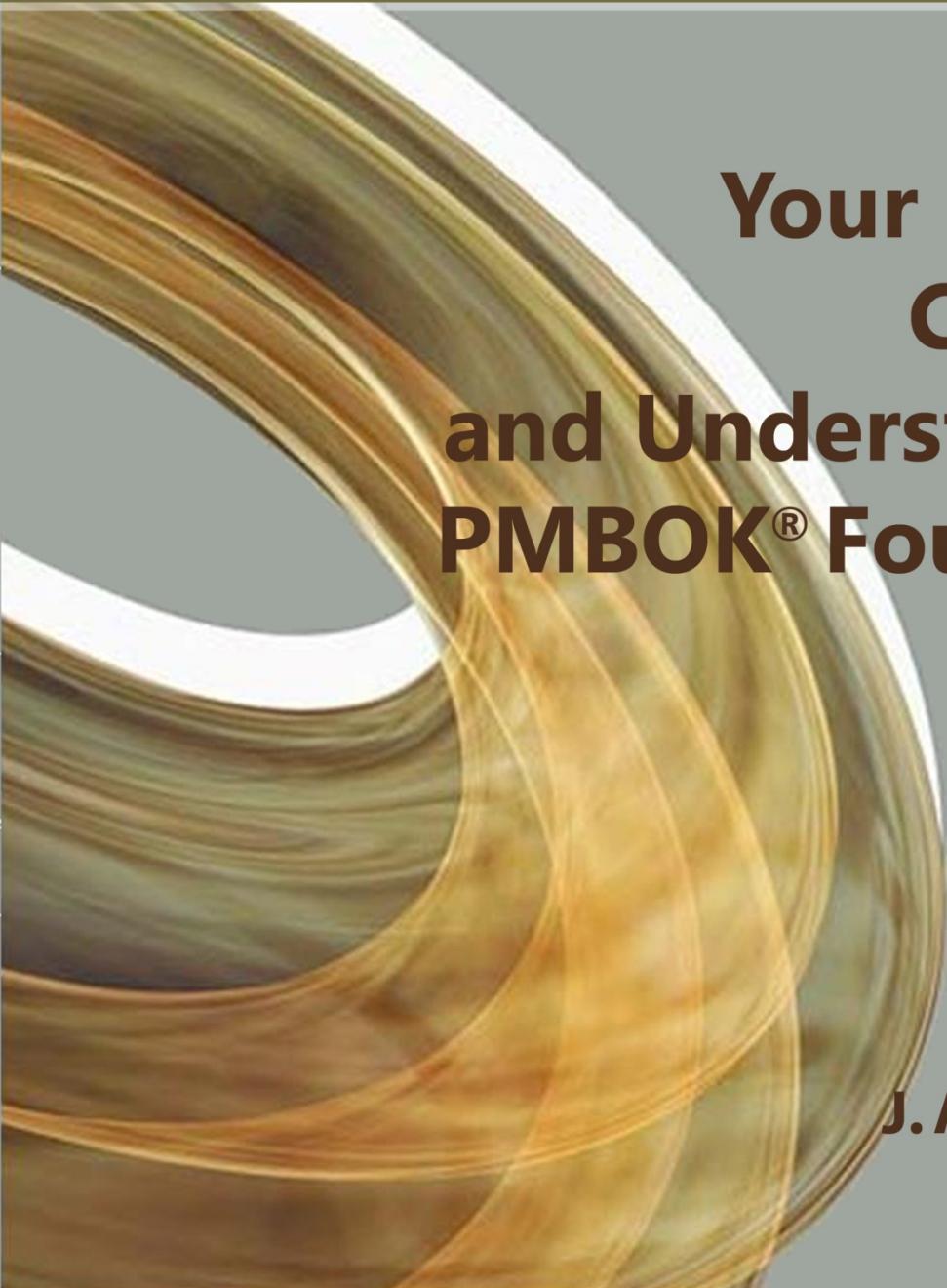


# **Project Management Road Trip®**

**For the Project Management  
Professional®**



**Your Key to PMP  
Certification  
and Understanding the  
PMBOK® Fourth Edition**

**J. Alex Sherrer, PMP**  
[www.PMRoadTrip.com](http://www.PMRoadTrip.com)

*Project Management Road Trip for the Project Management Professional:  
Your Key to PMP Certification and Understanding the PMBOK Fourth Edition*

**© Copyright 2009 by J. Alex Sherrer**

ALL RIGHTS RESERVED

Version 2010.04.18

Visit Project Management Road Trip on the Internet at <http://www.PMRoadTrip.com>

“PMI,” “PMP,” “Project Management Professional” and “PMBOK” are registered certification marks of the Project Management Institute, Inc. in the United States and other nations.

“Project Management Road Trip” is a registered trademark of Joel Alex Sherrer.

Although the author has made every effort to ensure accuracy and completeness of information contained in this book, he can assume no responsibility for errors, inaccuracies, omissions, or inconsistencies contained herein.

## TABLE OF CONTENTS

---

<b>5.0 Project Scope Management.....</b>	<b>95-126</b>
Overview .....	97
Product and project scopes .....	98
Requirements management plan.....	100
Scope management plan.....	101
5.1 Collect requirements .....	102
5.2 Define scope .....	107
5.3 Create WBS .....	113
5.4 Verify scope.....	118
5.5 Control scope.....	121
Chapter summary.....	124
Exam summary .....	125



## 5.0 Project scope management

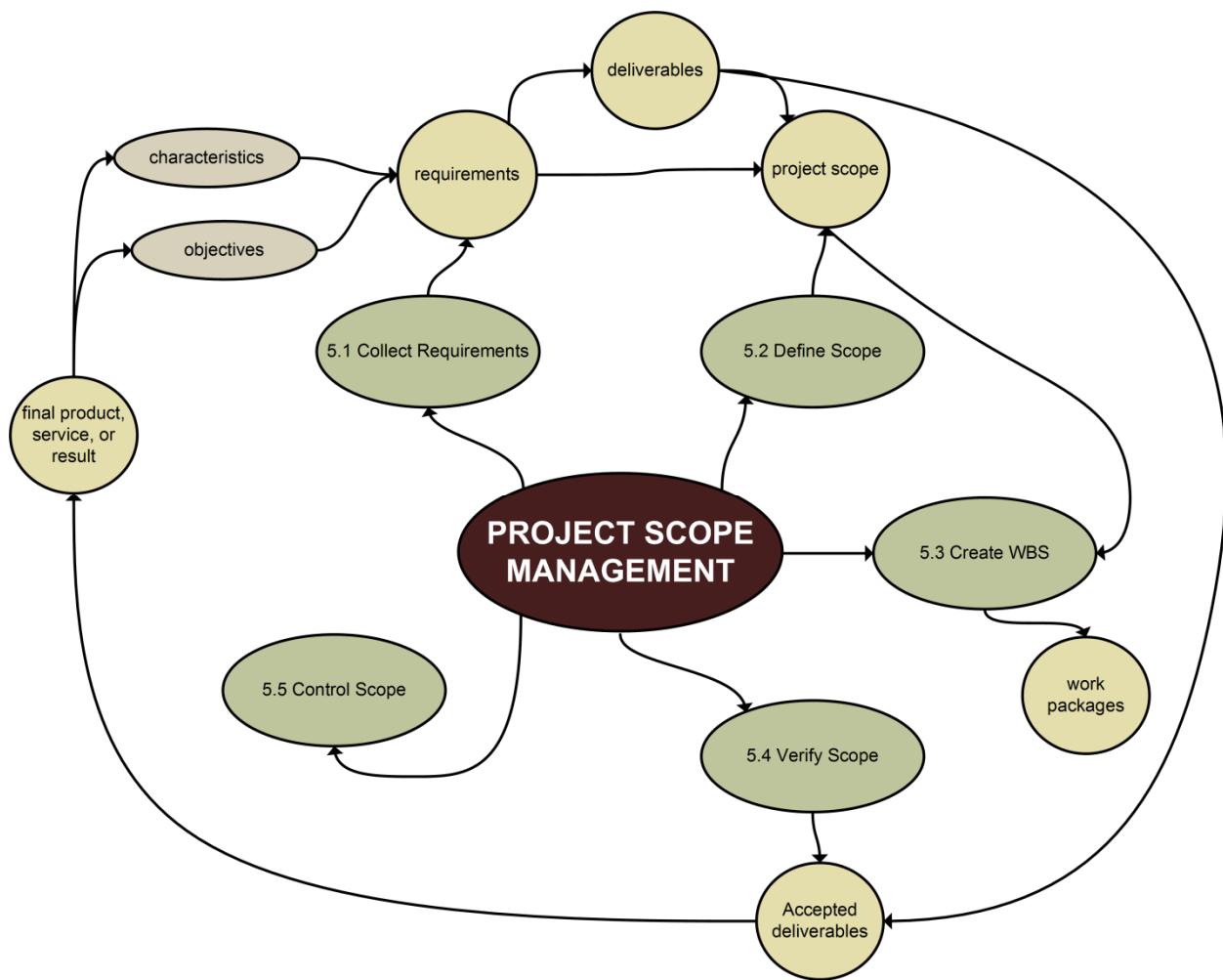
Chapter Difficulty

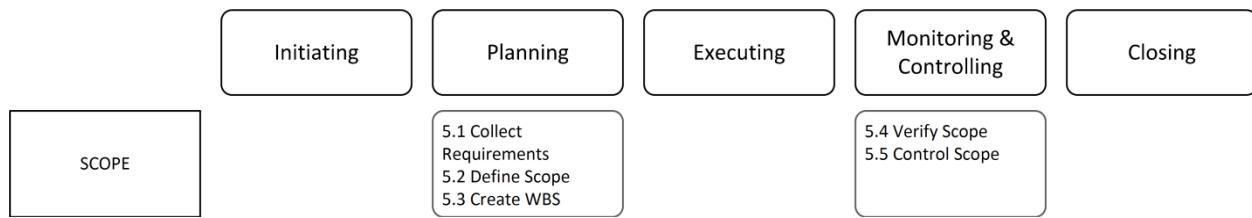
Easy

Moderate

Hard

Project scope management is concerned with the work that is needed to achieve the project's deliverables. It includes identifying the objectives and deliverables, converting objectives into action-oriented tasks, making sure that the project team does only the work agreed upon, and verifying that the completed work meets the requirements.



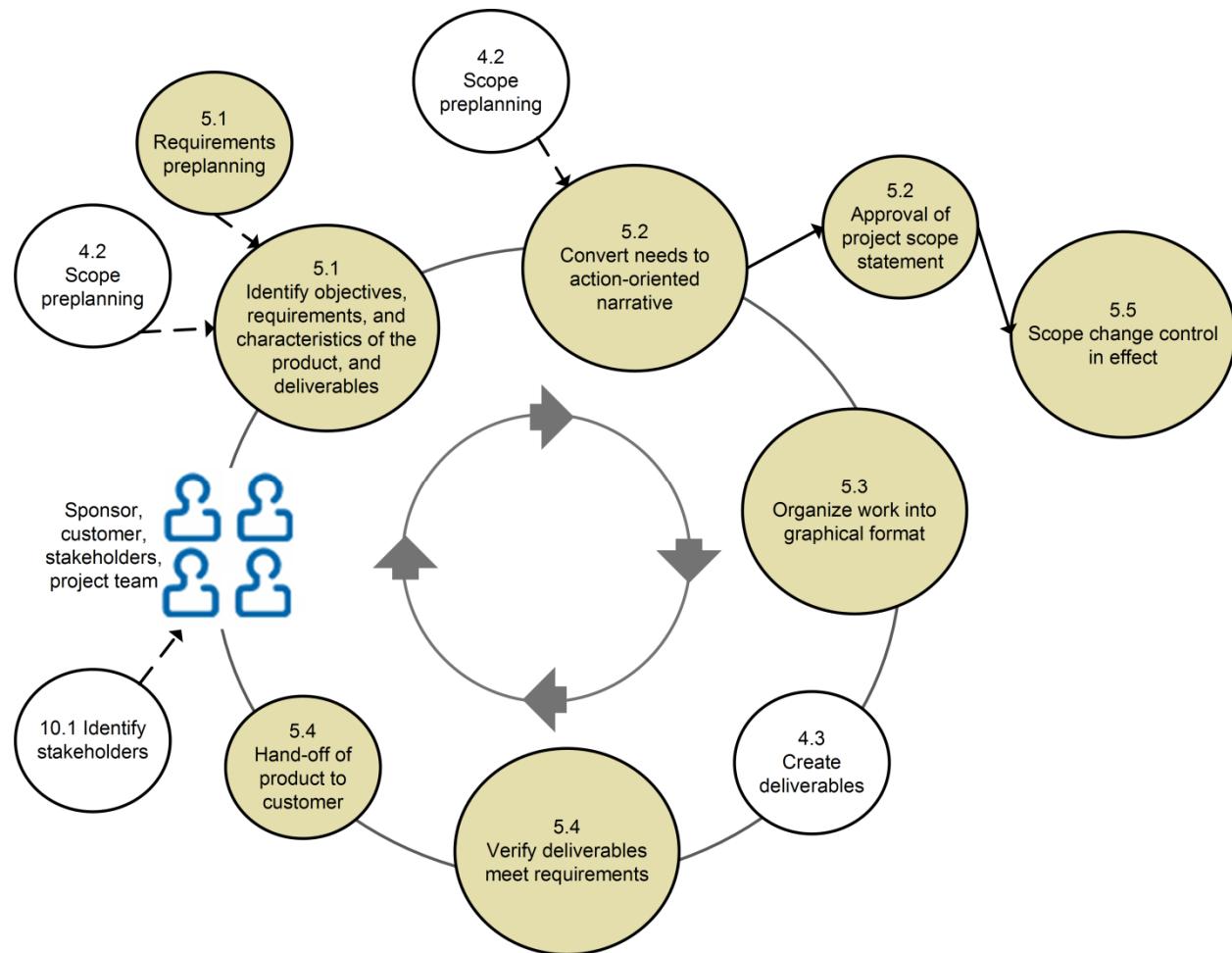


### Keywords

- Affinity diagram
- Alternatives identification
- Brainstorming
- Decomposition
- Delphi technique
- Dictatorship
- Gold-plating
- Idea/mind mapping
- Inspection
- Majority
- Nominal group technique
- Planning components
- Planning packages
- Plurality
- Product analysis
- Product scope
- Project scope management
- Project scope statement
- Prototype
- Requirements management plan
- Requirements traceability matrix
- Rolling wave planning
- Scope baseline
- Scope control
- Scope creep
- Scope management plan
- Scope verification
- Unanimity
- Variance analysis
- WBS components
- WBS dictionary
- Work breakdown structure
- Work packages

## Overview

**Project scope management** processes are concerned with the work needed to meet the project's objectives. These processes start immediately after the project is initiated and continue until the project is closed. Though they are easiest to follow in the logical order in which they're presented, scope management processes are iterative and will be revisited throughout the life of the project and may be performed several times in a multi-phased project.



- **5.1: Collect Requirements:** Determines what objectives and deliverables the project needs to meet.
- **5.2: Define Scope:** Converts the objectives and deliverables into action-oriented tasks in a narrative format.
- **5.3: Create WBS:** Organizes the project scope into a graphical categorization of deliverables and activities.
- **5.4: Verify Scope:** Ensures that the deliverables produce fulfill the agreed upon requirements.
- **5.5: Control Scope:** Makes sure that only the work agreed upon is performed by the project team.

There are four critical activities that should be completed before scope management can begin:

1. Since we can't determine the work to be performed without a general understanding of the project, we'll need the project charter (4.1, Develop Project Charter).
2. In order to determine the explicit objectives and deliverables the project needs to produce, we're going to have to talk to the customer and stakeholders, which means we need to know who all the stakeholders are. The stakeholder register, created in section 10.1 (Identify Stakeholders), provides us this list.
3. The specific needs of the customer and stakeholders will drive the project work (and thus the project scope), so before we start collecting those requirements, we'll need to know how we'll approach gathering, organizing, prioritizing, and managing of the requirements. All this is described in the requirements management plan, a subsidiary component of the project management plan. The requirements management plan is created in section 5.1 (Collect Requirements)
4. We'll need to pre-plan our approach towards identifying, categorizing, and controlling the project scope. Scope preplanning includes things like how the work will be organized, what approvals are required for the scope, and what kind of scope change control will be needed. All this information is contained in the scope management plan, created as a subsidiary component of the project management plan in section 4.2 (Develop Project Management Plan).

### Product and project scopes

Before we get too far into this knowledge area, let's revisit the product scope and the project scope. The two scopes are closely related yet serve distinct purposes, so understanding their differences will help us better grasp the scope management processes.

#### Product Scope

The **product scope** describes the characteristics of the project's final product, service, or result, and we can think of it as describing *what* we need to produce. It's a good project management practice to create a product scope if it doesn't exist. In generalized terms, the customer and stakeholders will be describing the *product* scope when describing what they need, so the project team can use that as a basis for authoring the product description.

Though it'll depend on what the final product, service, or result is and whether it's tangible or intangible, the product scope describes:

- What product, service, or result will do
- How the product, service, or result will be used
- How the product, service, or result will function
- What the product will look like, what the service is, or what the result will be

- What impact the product, service, or result will have on the organization, customer, stakeholders, and business processes
- Describes any constraints, restrictions, standards, regulations, and other requirements related to the product.

The project management team should know the product in great detail, and that the project team needs at least a solid understanding of it, and this holds true even if the project is only creating a portion of the overall product. The product scope should not be seen as a “need-to-know” item because the fact is that everyone –the entire team and the stakeholders—need to know what the project’s final product, service, or result is all about. When everyone understands the product scope the team is more likely to come up with more efficient, innovative, and resilient project decisions.

### Project Scope

The **project scope** textually describes the work the project team will perform to achieve all the project’s objectives. We can think of it as describing *how* we’ll produce the project’s deliverables, which not only includes the final product but any other deliverables required, such as those for project management activities. Because the project scope will drive what activities occur, it must describe in great detail and without any chance of misinterpretation how and in what steps the deliverables will be produced.

```
graph LR; A["Product Scope  
*Characteristics  
*Attributes  
*Purpose  
*Use  
*Specifications"] --> C((Work needed to accomplish)); B["Project objectives"] --> C; C --> D["Project Scope"]
```

The flowchart illustrates the process of defining project scope. It starts with two input boxes: 'Product Scope' (containing characteristics, attributes, purpose, use, and specifications) and 'Project objectives'. Both boxes point to a central circle labeled 'Work needed to accomplish'. This circle then points to a final box labeled 'Project Scope'.

Without a well-defined project scope and the controlling processes to protect it from unmanaged changes, the project is likely to encounter **scope creep** –changes that have a way of working themselves into a project that blow its schedule and budget.

## Requirements management plan

**Requirements** are the constituent needs, objectives, and specifications the project's deliverables must meet in order to be successful. Preplanning is needed to determine how requirements from the customer, stakeholders, and project team will be collected, organized, prioritized, tracked, and measured. The completed preplanning is documented in the **requirements management plan**, created in the Collect Requirements process (section 5.1) though it's a subsidiary component of the project management plan.

Discussing the many ways in which requirements can be gathered can end up in a heated and impractical debate. The short answer is that you should gather requirements using whichever method works for you. Whether you prefer a written document, screen diagrams, prototyping or use cases, the most important outcome is that the people who need to understand the requirements can do so.<sup>1</sup>

Martin Bauer, Managing Director of designIT, from Requirements Gathering Essentials

deliverables that go into the product's final make up.

How detailed the requirements management plan needs to be depends upon the complexity of the project and the number of requirements that are expected surface, but in general the plan should cover:

- The kinds of interactions that will occur with the customer and stakeholders to collect requirements
- The categorization methods for requirements. The categories will be specific to the industry, product type, development life cycle, and application area of the project, but examples of classifications might include functional, customer, design, quality, and testing.
- With the anticipation that there will be more requirements than can be met within the project's schedule or budget, the requirements management plan should describe the prioritization, scoring, and approval methods that will be used to develop an approved requirements list.
- The configuration management activities that will ensure how changes to requirements will be evaluated, tracked, approved, and reported.

### Scope management plan

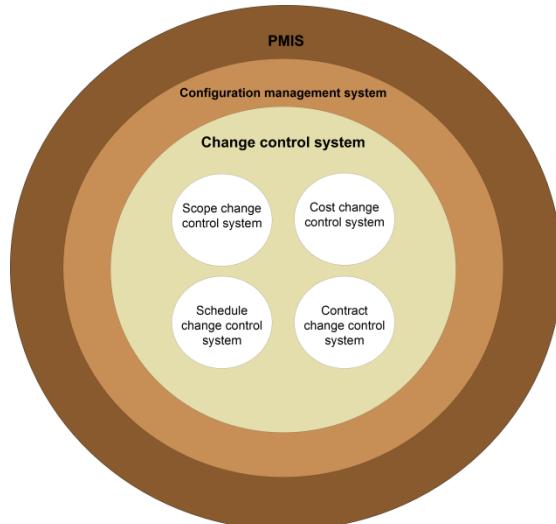
Each knowledge area has at least one subsidiary plan focusing on a specific subject as part of the overall project management plan. Preplanning is the purpose of these components, and these plans map out the specific requirements for the deliverables and project management processes that will take place in that knowledge area. This preplanning may sound like a lot of work, but we can think of these subsidiary plans as being the scope statements for the knowledge area because they describe the who, what, where, why, and how of the project management work that will be performed for that section's subject matter.

The **scope management plan** is a subsidiary component of the project management plan, created in the Develop Project Management Plan process (4.2), and it should be established before too much work begins in the scope processes because it describes:

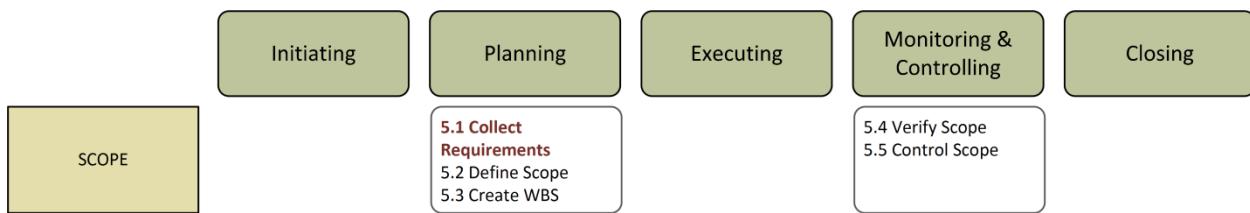
- What activities, methods, and processes will be used for creating the project scope statement
- How the scope will be approved by the stakeholders and project team
- How the project work will be categorized and organized (the work breakdown structure)
- How likely scope changes are to occur on the project
- What the procedure will be for the stakeholders to formally accept the project's deliverables.

As with other subsidiary plans, templates can help provide the basis for the scope management plan. The complexity of the project and how likely scope changes are to occur will be factors in deciding how detailed the scope management plan needs to be.

The procedures for scope changes are addressed by the project's change control system, but some projects may need a rigorous change control system that focuses specifically on scope impacting the scope. If so, the scope management system will address how scope changes are logged, evaluated, documented, approved or declined, and how approved scope changes will be handled. The scope change control system is still part of Integrated Change Control (section 4.5) and so is still part of the project's change control system.



## 5.1 Collect requirements



The expectations, needs, and objectives of the customer and stakeholders are identified, clarified, documented, prioritized, and managed through activities the **Collect Requirements** process. The success of the project is directly tied to how well these activities are performed because requirements are what drive the project's scope, which is central to all project management activities.



### Eliciting requirements

The project charter helps by providing a starting reference for the initial requirements because it contains the project objectives and statement of work, but it shouldn't be relied upon as the only source. There are many different methods for eliciting requirements, but they all center around interactions with the sponsor, customer, and stakeholders, and most projects will need a combination of different methods performed several times. In practice, the techniques used to define the product scope, requirements, and the project scope are often performed simultaneously since all are closely related.

Passive observation of how people perform their processes can be helpful as can “participant observation” by actually performing a process or procedure to see how it’s done. Interviews, brainstorming, focus groups, and workshops will give forums for the stakeholders to express themselves, and skilled project managers, team leaders, and business analysts will view themselves as liaisons between the stakeholders and project teams and rely on good communication and listening skills to convert expressed needs into project requirements.

Regardless of the type of product, service, or result the project is developing, collecting requirements is a collaborative effort with a lot of clarification, translation, and restatement of requirements between all parties.

Collecting requirements is much like searching for a treasure chest. Most requirements will start out being identified only at a very broad level, but a good analyst will collaboratively follow that trail with the stakeholders, digging deeper and deeper, until he or she has explicit, well-defined jewels we call requirements.

Our purpose isn't so much to lead the discussions during these meeting but rather to serve as a facilitator:

- Restate what was said. This is helpful not only to make sure that we've grasped the speaker's meaning, but it can also help others in the group better understand by hearing it phrased differently.
- Avoid acronyms and jargon unless we're absolutely sure everyone understands exactly what the terms mean.
- Withhold judgment while another is talking.
- Ask probing and direct questions that require more than simple yes or no responses, and give the person time to respond in his or her own manner.
- Make sure everyone has an opportunity to provide his or her thoughts.
- Listen fully to what everyone is saying, and watch non-verbal cues. If we suspect someone isn't agreeing with what is being said, give him or her an opportunity to discuss why. It's better to bring these issues out in the open rather than have somebody disagree later after everything has been approved.

### Prioritizing requirements

In all likelihood, there will be more requirements found than what can be accomplished within the project's constraints. Time, money, and resources will limit what requirements can be accepted, so a method of prioritizing and agreeing upon requirements should have been defined in the requirements management plan.

When only a finite amount of requirements can be addressed, prioritization can often bring out the worst qualities in people since everyone will have a different view of which requirements are mandatory, which are essential, and which are merely desirable. The group may need different decision making techniques for different categories of requirements, but in some manner the requirements will be decided by:

- **Unanimity:** Every single group member agrees on the same option
- **Majority:** The option with more than 50% of support from group is chosen
- **Plurality:** The option with the largest block of support is chosen (even if the largest block doesn't have a majority)
- **Dictatorship:** One person chooses the option.

There are many methods and tools that can help with the prioritization process by introducing some objectivity into the decision-making process. Scoring matrices, decision trees, and voting methods are very frequently used.

Two easy voting methods are:

Numerical Assignment Technique (Joachim Karlsson<sup>6</sup>)

Participants are asked to rank requirements on a scale of 1 to 5 indicating the importance of the requirement. The final ranking of requirements is the average of all responses. The scale is:

- 1: The customer does not need it
- 2: The customer would accept its absence
- 3: The customer would appreciate it
- 4: The customer doesn't want the product without it
- 5: The customer will not accept the product without it

100-Point Method (Leffingwell and Widrig<sup>7</sup>)

Each stakeholder is given 100 points that he or she can use to vote for any number of requirements. There are no limitations, so a voter could put all 100 points on a single requirement or split his or her votes in any manner.

### Requirements traceability matrix

Projects often have several different types of requirements. There'll be requirements needed:

- to achieve the business objectives
- to create the deliverables
- to conform to product design or development specifications
- to meet project objectives
- to meet project management needs
- to develop testing strategies for deliverables.

Requirements are documented in a method that's specific to the project. For example, some projects may rely on textual descriptions, use cases, and process maps while another project may require schematics. But in addition to documentation, requirements are logged onto a **requirements traceability matrix**. The main purpose of the matrix is to enable every requirement to be logged and attached to a project objective or to another requirement in a hierarchical manner. This makes sure that every requirement adds value that the customer and stakeholders need, and it exposes the relationships requirements have with each other. Knowing the relationship between requirements makes sure that changes can be accurately evaluated for whether they support the project's objectives.

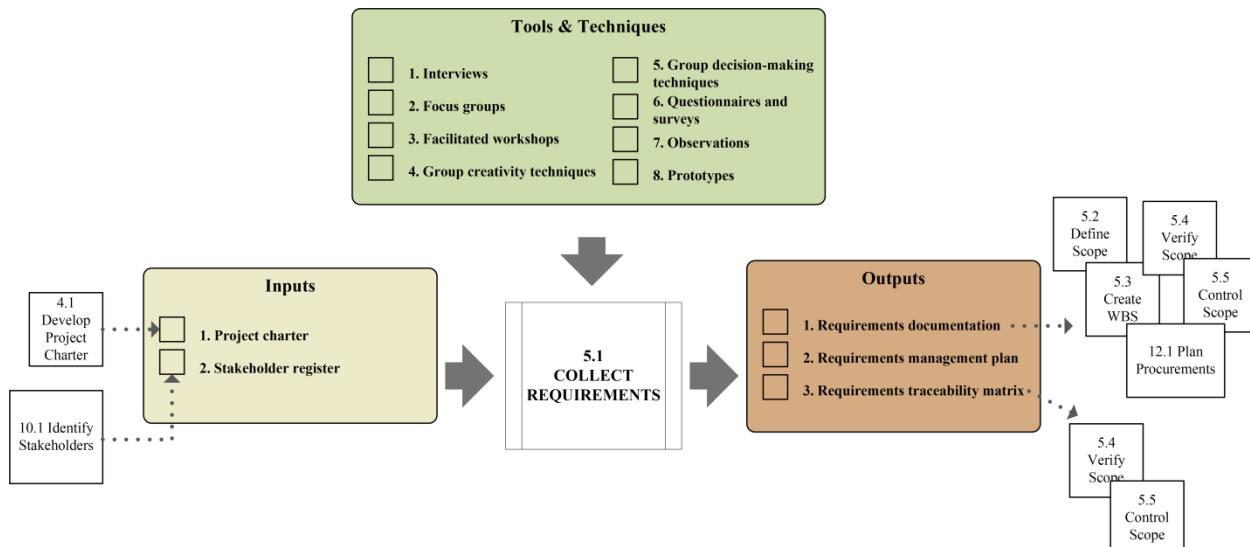
The exact form the matrix will take depends primarily on the categorization method defined in the requirements management plan. But as an example, let's pretend that the customer has a requirement that a signature is required from the credit card holder only for transactions over \$25. This single requirement will spawn some related requirements that will likely have to do with the functionality of the point-of-sale terminal and some testing requirements for the quality personnel on the project.

Type	Identifier	Description	Linked To
User	U100	Signatures are required only for sales greater than \$25	
System	S155	The electronic signature pad will not require the customer to sign for credit card transitions of \$25 or less.	U100
Testing	T250	The signature pad will not stop the sale for a signature if the total sale <= \$25.	U100

To be useful, at a minimum, the requirements traceability matrix should contain:

- A unique identifier for the requirement
- A textual narrative of the requirement
- The requestor and owner of the requirement
- The requirement's priority
- Relationship to other requirements
- Requirement's status (whether under review, approved, canceled, deferred, and so on)
- References to other documentation about the requirement.

## 5.1 Process decomposition



### Inputs

#### **Project charter**

The project charter formally authorizes the project and the project manager. It also provides the business case, objectives, and success criteria of the project. The charter provides the framework for project planning activities.

#### **Stakeholder register**

The stakeholder register identifies all project stakeholders and contains attributes such as the person's name, title, position, project interest, expectations, and influence.

### Tools and Techniques

#### **Interviews**

Interviews are any formal or informal interaction technique involving questions.

#### **Focus groups**

Focus groups are moderated discussions with a group of people usually conducted informally and centered on well-defined subjects.

**Facilitated workshops**

Facilitated workshops are interactive discussions with participants usually conducted informally to encourage an open dialogue. Facilitated workshops are central to many different methodologies, and include Quality Function Deployment, Voice of the Customer, and Joint Application Development.

**Group creativity techniques**

These techniques encourage lateral thinking for creative problem-solving or identifying innovative solutions. They include brainstorming, idea and mind mapping, affinity diagrams, and nominal group technique.

**Group decision making techniques**

These are group techniques for reaching a decision. At the root they involve decisions based on unanimity, majority, plurality, or dictatorship.

**Questionnaires and surveys**

These are paper- or electronic-based questioning to collect responses from a group of people. These are useful for obtaining a large number of results relatively quickly.

**Observations**

These are interactive participation or passive observation to collect data, such as viewing how a process is performed. It's useful for defining and measuring existing processes, identifying improvements, and collecting requirements.

**Prototypes**

Prototypes are models or designs of a product, usually with limited functionality. They're useful for providing a tangible example of a concept or idea so that others can provide feedback.

### **Outputs**

**Requirements documentation**

The product scope and project objectives are broken down into requirements and described in a collection of requirements documentation that's applicable to the project and requirement type. Requirements can be documented and described in a number of different levels (executive, summary, and detailed) and in a variety of methods (textually or visually).

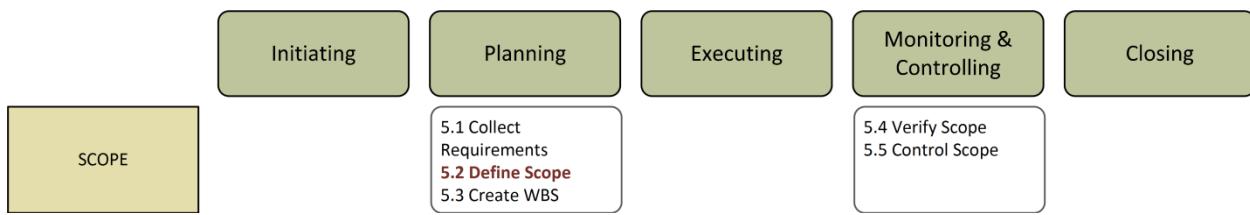
**Requirements management plan**

The requirements management plan describes how project requirements from the customer, stakeholders, and project team will be collected, organized, prioritized, tracked, and measured. It is a subsidiary component of the project management plan.

**Requirements traceability matrix**

All requirements are logged onto a requirements traceability matrix. The main purpose of the matrix is to enable every requirement to be logged and attached to a project objective or to another requirement in a hierarchical manner.

## 5.2 Define scope



The project scope is established through the Define Scope process. In most cases, the product scope will also be created through activities in this process or undergo revisions. The **project scope statement** explicitly describes all the project's deliverables and the work necessary in order for the project to meet its objectives and all its approved requirements.

## project scope statement

<b>deliverables</b>	<b>milestones</b>
<b>assumptions</b>	<b>risks</b>
<b>constraints</b>	<b>exclusions</b>

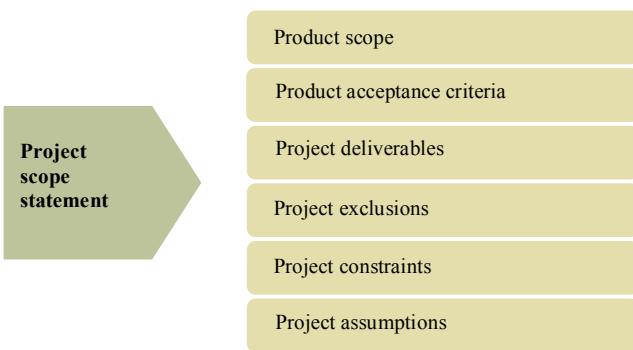
**alternatives identification**

## product analysis requirements product scope

**DEFINE  
SCOPE**

The scope statement will either include or make references to project constraints, assumptions, and any scope exclusions. The scope also describes the criteria the final product, service, or result must meet and the process for the customer's acceptance of the product.

The scope statement needs to contain or reference other documents that describe all these elements:



- **Product scope:** The characteristics of the product, service, or result for which the project was undertaken. In projects that are part of a larger program, the project itself may only be creating components of the product, but the product scope or product description is still necessary so that everyone knows what the overall objective is.

- **Project objectives:** Objectives are the success metric for the project. Specifically, what will it take for the project to be considered successful? This includes the business, cost, schedule, technical, and quality objectives, and other specific targets should be included where applicable.
- **Project requirements:** The capabilities that the product, service, or result must possess and meet. Requirements are the translated expectations and needs of the stakeholders into prioritized, descriptive requirements and work items.

- **Project exclusions:** Nearly just as important as what IS in the project, the scope should include items that are excluded from the project. Doing this helps eliminate any confusion within the stakeholders or project team.
- **Project deliverables:** The core product, service, or result should be fully described, as well as any ancillary deliverables. Any needed project artifacts, those documents not directly related to the deliverables, such as management, technical, or status reports, should also be described.
- **Product acceptance criteria:** The process and criteria for product acceptance should be defined. This includes customer-specific requirements and any testing or other threshold limits.
- **Project constraints:** Any limiting factors the project must work within, such as deadlines, budget, staffing, facilities, equipment, materials, or contractual restraints, should be described.
- **Project assumptions:** Every project has assumptions, and these should be described because assumptions are risk factors.
- **Risks:** Risks should be identified at least at a high level. The risk register is where all risks are logged, but having major risks explained in the scope statement helps make everyone aware and on the lookout for them.
- **Milestones:** Any important dates, including deliverable- or artifact-oriented dates should be included in the project scope statement.
- **Approval requirements:** Any specific approval requirements for items such as deliverables, documents, and work should be described.

Rob McGannon, founder of MINDAVATION, wrote that milestones should not be looked upon only as deliverable-oriented events. Especially on large projects, stakeholders may lose interest when they can't see the light at the end of the tunnel. One way of keeping stakeholders engaged is to create shorter-term milestones and success points that are directly tied to the business purpose for the project.<sup>2</sup>

Bob McGannon, PMP, Founder of MINDAVATION

The project scope statement contains a lot of information because it is what drives every single activity performed on the project. *If something isn't described within the detailed project scope statement then the work should not be done or the scope statement needs revised to include the work.* Because of its level of detail, the project scope will require several iterations before it's finalized, and even then it's highly susceptible to changes.

The project scope should be understood well by the project team since their activities are derived from it, but the project scope statement also serves the project stakeholders by being a reference point for what the project is all about. The scope isn't acceptable if it's so technical that only a handful of people on the project can understand it. That only invites conflict later in the project as stakeholders are likely to feel they were led into agreeing to items they didn't fully comprehend. So the scope statement should be written in a manner that makes it clear to everyone. Sometimes this can be difficult, especially when the project involves requirements that are very technical and the stakeholders and project team have differing degrees of technical knowledge. One way of doing this is to tailor different sections of the scope document to the different audiences while still maintaining the integrity of the project scope statement, such as keeping overly technical requirements that may be needed by a small portion of the project team in a document that is clearly referenced by the scope statement (and also referenced by the WBS dictionary).

Another item for us to watch for is gold-plating. **Gold-plating** is when requirements or deliverables are added to the project scope that are not expressly requested by the customer or project --for instance,

adding higher-end appliances to the scope as part of a remodel project when mid-range appliances would've been acceptable. But gold-plating can be even subtle elements, like minor software functionality that wasn't requested. We all want to please the customer, but gold-plating is a form of scope creep, and the customer is best served by providing them exactly what it needs. If we think the deliverable or requirement would be helpful for the customer or stakeholders, we should bring it up for discussion --not just add to the scope because we think the customer would appreciate it. The customer is more likely to appreciate being consulted about it rather than paying for something it didn't ask for.

It's acceptable for portions of the project's deliverables to be remain somewhat undefined. For example, components that are required later in the project may not be fully defined up front, but rather the full requirements for them will be developed in subsequent planning sessions. This is called **rolling wave planning (RWP)** --items not required immediately by the project are planned for later as the need for them draws closer. Rolling wave planning is not an excuse for large "black holes" in the scope or other project planning documents. The overall characteristics of the component should still be described as fully as possible, and activities should be put into the project schedule for further planning on these items.

It might seem that agile software development and project planning are mutually exclusive: how can you plan future release dates if the one certainty is that those dates (and the scope) are going to change? However, planning is actually an integral part of agile development. It isn't even all that different from "normal" project planning, when you get down to the fundamentals. The main difference is that with agile planning (also known variously as adaptive planning and rolling wave planning), you plan ahead in broad brush strokes – release milestones – and only plan in detail for the next iteration.<sup>3</sup>

Matt Stephens, Grappling with Eels - First Wriggle: The Slippery World of Agile Planning

## Product analysis

A project scope can't be developed without having a good understanding of the product, even if our project is only delivering a part of a larger product, service, or result. Disconnects between the product scope and project scope can result in deliverables that don't fulfill the need for which they were intended. **Product analysis** is a tool that translates product objectives into tangible project requirements and deliverables by dissecting the product into components.

Some of the generalized methods of product analysis include:

- Use cases/modeling: These are step-by-step accounts of system behavior and how it interacts with the user. These are very common in software development projects, but they are also widely used in other industries.
- User scenarios/user stories: These are also step-by-step accounts, but they take a more fictional narrative approach, and they are usually written by the customer, user, or analyst and they describe interactions less from a system's standpoint as they do from what the customer needs the product to do.
- Product breakdown: This approach takes a complex product and separates it into simpler components that can be more easily described.
- Functional analysis/functional requirements: These approaches take an even more detailed approach than use cases, and describe in detail specific functions of the product from the system's viewpoint.

- Systems engineering: This approach is highly different from industry to industry, but it generally takes a broad, overall view of the product life cycle, focusing on balancing the customer's needs, cost, quality, and complexity.
- Quality function: A quality management practice whose goal is to fully understand the customer's needs.

### Alternatives identification

There is usually more than one way to solve a problem or reach our project's objectives. **Alternatives identification** looks for different approaches in performing the work required by the project or in the methods for achieving the project's objectives. This can result in project and organizational benefits, such as less production costs or time, better quality, or competitive advantages. Alternatives identification can also turn up methods of addressing stakeholder concerns, usually negative, while still maintaining the integrity of the project's objective. Looking for alternative solutions requires energy, effort, and creating an environment where this behavior is encouraged.

Project managers may need to lead and encourage stakeholders and the project team through creative thinking techniques. We're speaking here about technical creativity (new techniques, technologies, processes, and procedures) rather than artistic creativity, though that form of creativity can also be applicable to some projects. There are two main branches of technical creativity: programmed thinking and lateral thinking. Programmed thinking is based on pattern recognition, and humans are especially good at this. We're able to recognize images, pictures, objects, situations, and learn by cause-and-effect. This type of thinking can help us refine and improve our work processes, but it generally leads only to small, incremental improvements. And the very fact that we rely on past experiences in programmed thinking can get us into a rut by approaching problems the same old way.

The need to be right all the time is the biggest bar to new ideas. It is better to have enough ideas for some of them to be wrong than to be always right by having no ideas at all.<sup>4</sup>

Edward de Bono, lateral thinking pioneer

Lateral thinking doesn't completely disregard programmed thinking, but it uses tools that help us break up the patterns and see things differently. Techniques such as brainstorming, free association, and mind mapping are lateral thinking techniques. The results of these techniques are a large pool of ideas which can later be reviewed more thoroughly. Regardless of the lateral thinking technique used, there are general guidelines that we can use to encourage the process:

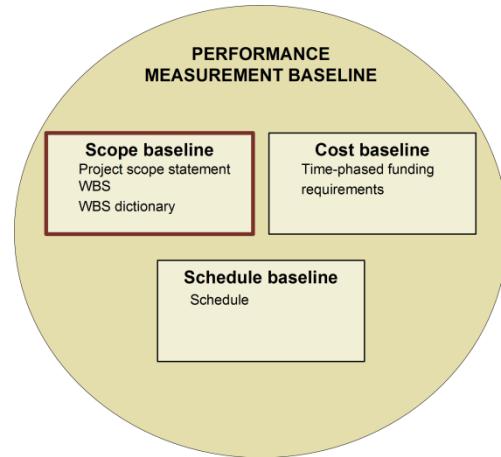
- Creative thinking is not an inherent talent. It can be learned and improved upon by anyone.
- Creative thinking requires us to believe we are creative (self-perception), so encourage, reinforce, and reward creative thinking in the project team.
- Creative thinking is usually enhanced through a group setting rather than an as individual activity.

- Do not criticize or judge any initial ideas. All ideas, even the most silly and absurd, should be welcomed.
- In the initial stages, quantity is more important than quality.

Engaging the project team and stakeholders in creative thinking techniques can also have rewards beyond alternatives identification. It can bond the group together, and help create an environment that encourages innovative thinking which is useful in all stages of the project.

### Scope baseline

Once the scope is approved and the WBS and WBS dictionary have been created (in section 5.3), the scope baseline comes into existence. The **scope baseline** is the collection of the approved project scope statement, the work breakdown structure, and the WBS dictionary. All project baselines are the current plans that have all approved changes incorporated into them. Once these documents are approved in some manner, the baseline is created, which just means that any further changes must go through the change control process. Baselines are *not* the original plans. Though this may seem illogical at first, an example may help us better understand baselines.

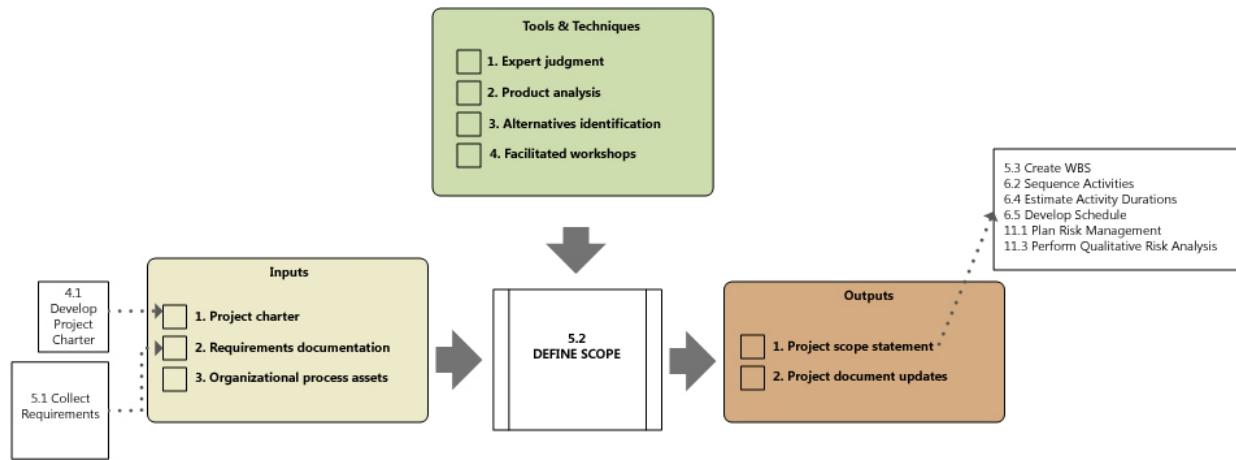


*Imagine that we originally planned for a low-budget surprise birthday party for a friend and had invited five other people. It was going to be held in our home and everyone was going to chip in and bring food. The entire event was only going to take two days to plan and cost no more than \$100.*

*But later everyone agrees that the guest of honor deserves a bigger party, and so it's now going to be held at a restaurant with dinner and drinks for 50 guests. The party is going to take us seven days to organize and will cost \$5,000. The big day arrives and everything comes off exactly as planned.*

Was our project successful? If we look at the original plan compared to what actually occurred, it took us 350% longer to execute the project and we exceeded our original budget by 500 times! However, the original plan is not what our comparison should be based on because the scope of the project changed midway through. Our comparison should be based on the seven-day schedule and budget of \$5,000. The purpose of any project management baseline is to provide measurements off the approved plans. We want what's currently been agreed upon as our measuring stick, and when we do this with the example, since the original plan was altered and agreed upon by all the stakeholders and we performed exactly as we expected, the project should be considered a success.

## 5.2 Process decomposition



### Inputs

#### Project charter

The project charter formally authorizes the project and the project manager. It also provides the business case, objectives, and success criteria of the project. The charter provides the framework for the project's requirements.

#### Requirements documentation

The product scope and project objectives are broken down into requirements and described in a collection of requirements documentation that's applicable to the project and requirement type. Requirements can be documented and described in a number of different levels (executive, summary, and detailed) and in a variety of methods (textually or visually).

#### Organizational process assets

Organizational process assets are the source of existing policies, processes, organizational data and knowledge. These assets include the entire collection of formal and informal methodologies, policies, procedures, plans, and guidelines, as well as the organization's "knowledge base," which includes historical performance data, labor information, service and maintenance history, issue and defect history, project files, and financial data. In developing requirements and the project scope, templates, checklists, and lessons learned from previous projects are useful.

### Tools and Techniques

#### Expert judgment

Expert judgment is based upon the experience and knowledge of subject matter experts. It's used to assess and evaluate the inputs and the information they contain.

#### Product analysis

Product analysis translates product objectives into tangible project requirements and deliverables by dissecting the product into components.

**Alternatives identification**

Finding alternatives looks for different approaches in performing the work required by the project or in the methods for achieving the project's objectives. Alternatives identification is applicable to many project components.

**Facilitated workshops**

Facilitated workshops are interactive discussions with participants usually conducted informally to encourage an open dialogue. Facilitated workshops are central to many different methodologies, and include Quality Function Deployment, Voice of the Customer, and Joint Application Development.

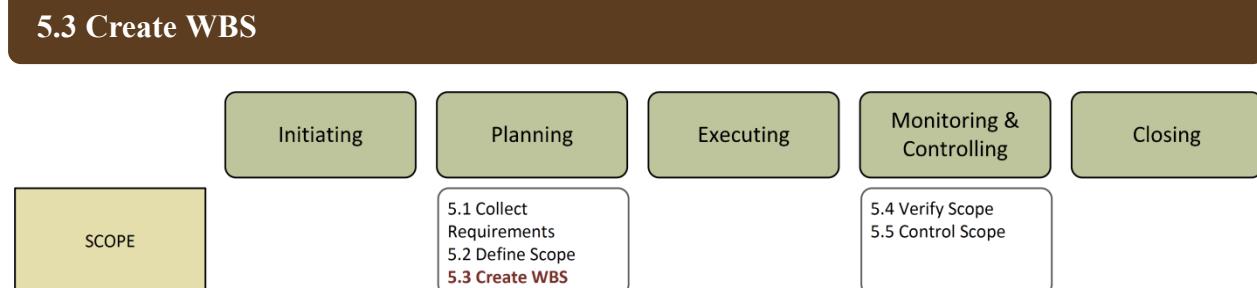
### Outputs

**Project scope statement**

The project scope statement details the measurable goals, objectives, deliverables, and requirements of the project, and what the acceptance criteria of deliverables will be. It also describes the work required to meet all objectives and deliverables of the project, and it also contains milestones, assumptions, risks, and costs. The deliverables will be compared to the project scope statement to make sure they comply with it.

**Project document updates**

Defining the project scope is likely to uncover additional requirements, so updates to requirements documentation and the requirements traceability matrix are to be expected.

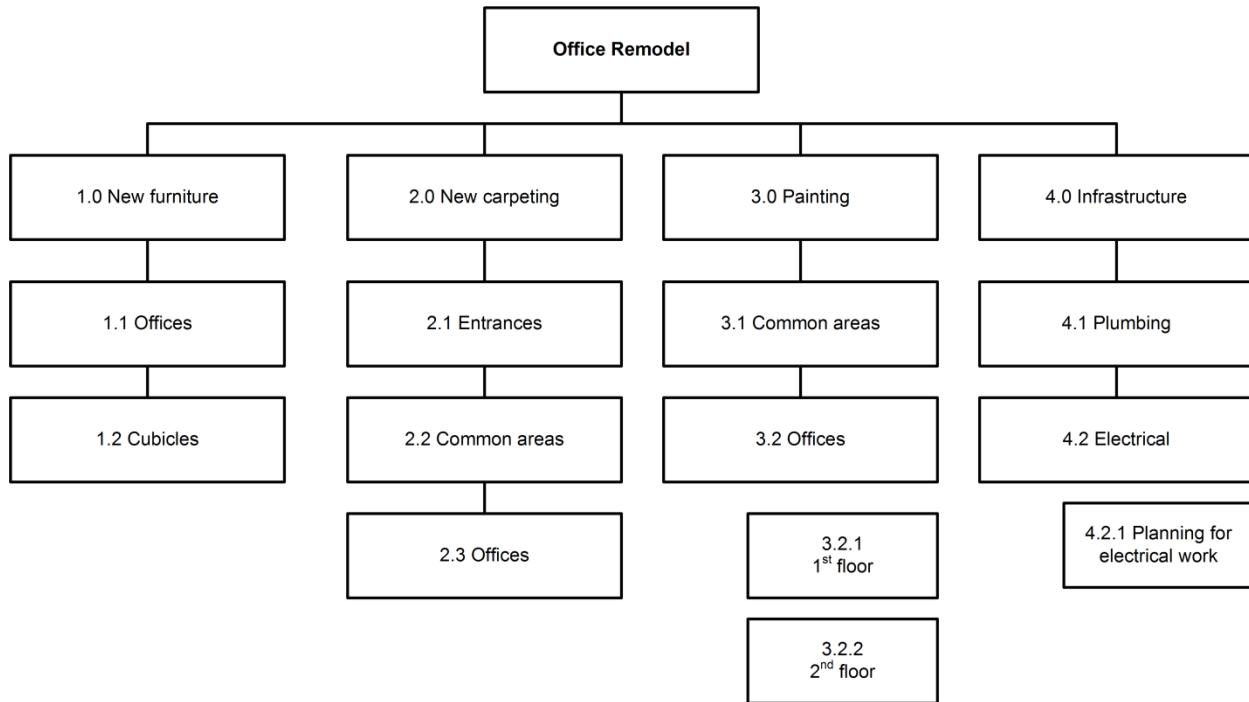


The **work breakdown structure (WBS)** is a deliverables-oriented, graphical, hierarchical representation of the work required to fulfill the project scope statement. It is somewhat like a genealogical family tree for the deliverables and requirements, and it's one of the most important project management tools.



A good work breakdown structure is a strong framework for good project management. The WBS serves several purposes:

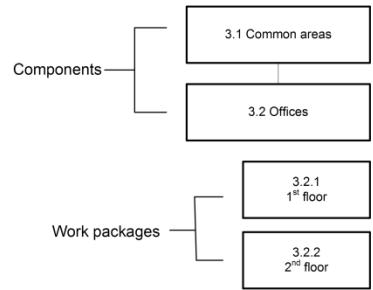
- Promotes better overall integration between the project management processes.
- It subdivides the work into manageable components that can be scheduled, estimated, and assigned.
- Through the process of creating and updating the WBS, it helps to identify needed work that might otherwise not have been discovered until later.
- It can be used as a visual communication tool for the customer, stakeholders, and project team.
- The WBS is an input to activity definition, cost estimating, cost budgeting, resource planning, and risk management planning.



All the work on the project must be accounted for in the WBS, including any project management activities. The lower levels of the WBS must roll up to the higher levels so that nothing is left out and no extra work is contained in the WBS that isn't described in the scope. This is sometimes referred to as the **100% rule**.

Templates and work breakdown structures from similar projects can provide a basis for our project's WBS. But how the WBS is organized depends on the industry, application area, and personal or organizational preferences, and though often shown as an organizational chart, outlines, fishbone diagrams, and other methods can be used. In whatever manner it's organized, the WBS should be organized by control accounts, which should correspond to deliverables. Control accounts are management levels in the WBS where scope, cost, and resource information can be summarized and used later for earned value performance measurements.

The WBS consists of **components**, which are the major nodes of the hierarchy and represent the deliverables or major work classifications, and **work packages**, which are the lowest levels of work within the WBS that can be easily managed.



Work packages should be small enough so that they can be assigned to someone, be scheduled, have a cost reasonably estimated, and be able to be managed and controlled. Work packages are created through **decomposition** --converting the deliverables into the work required necessary to create them, and breaking the work down into smaller and smaller increments until it's manageable. For example, in a multi-floor office building remodel, a work package to "Paint Offices" would be too broad. A better decomposition would be to break it up by floor or even further depending upon how many offices are involved.

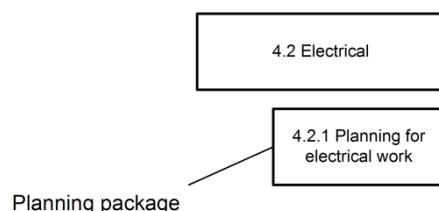
An accurate WBS is fundamental to project success. Creating it reveals how a project's major outcomes are supported by key objectives or strategies, and in turn, what tactics or activities are needed to implement these strategies. No matter what conceptual model you use to help you understand the hierarchy of the work, the list of activities you end up with becomes the backbone of all other planning, monitoring, and controlling decisions you make.<sup>5</sup>

Ron Black, *The Complete Idiot's Guide to Project Management with Microsoft Project 2003*

There is no hard-and-fast rule for what is an adequate level of work package decomposition. One general heuristic is that a work package should take no fewer than eight hours of effort and no more than 80 hours. One thing for us to keep in mind is that the WBS is not a task list. It's not intended to provide significant detail about the activities. In fact, excessively decomposing work packages can have a detrimental effect on the WBS by making it overly complex, confusing to view, and difficult to maintain.

It may not be possible to fully decompose all deliverables because the planning for them will occur later in the project. Rolling wave planning is an acceptable technique in which future project activities are not fully planned for up front. But these activities still need accounted for in the WBS, so planning components and planning packages will be used as placeholders to show that additional planning is necessary.

A **planning component** is a high-level node that indicates further planning is necessary. **Planning packages** are similar to work packages in that they're placed below planning components, and they are the node that holds the planning activities, for example "Planning for electrical work." Just like a work package, the planning package should be assignable to a resource, be able to have its planning activities scheduled, and be able to have its effort reasonably estimated.



Decomposition may also not be possible when the deliverable or a portion of it is contracted out to a party outside the project. In this case, a node is placed within the WBS, but it will likely be only at a high level and not contain any work packages.

### Steps to Create a WBS

1. Identify all the project's deliverables and artifacts. Keep the list textual at this stage rather than graphical. Level zero of the WBS will be the project.
2. Look for any inherent categorization and rearrange the list as necessary to organize it under main categories. Strive hard to find a categorization that is related to the deliverables. These categories will be the first-level hierarchy of the WBS.
3. Don't forget to include project management or other related activities. These are also requirements of the project.
4. Decompose the deliverables down into the major activities needed to produce them. Just as we did in step 2, we might begin seeing a categorization emerge from individual activities that will indicate additional components ("sub-components") that work packages can be grouped under.
5. It's likely we'll find some general dependencies and natural sequencing of activities while we're decomposing. We'll want to reflect these as best as we can at this stage, but don't get too mired down in these details because dependencies and sequences will be addressed while we're creating the project schedule (chapter 6).
6. Continue decomposing until we have a level of work packages that are applicable to our project, and that can be assigned, scheduled, estimated, and managed. If we're unable to decompose something to a small enough detail, we may need to insert a planning component or package until we can better define the deliverable.
7. Review our list of deliverables, components, and work packages. If the classification and organization isn't clear, regroup until it is.
8. Create the graphical WBS from the textual lists. Review it for clarity and visual aesthetics.

### **WBS dictionary**

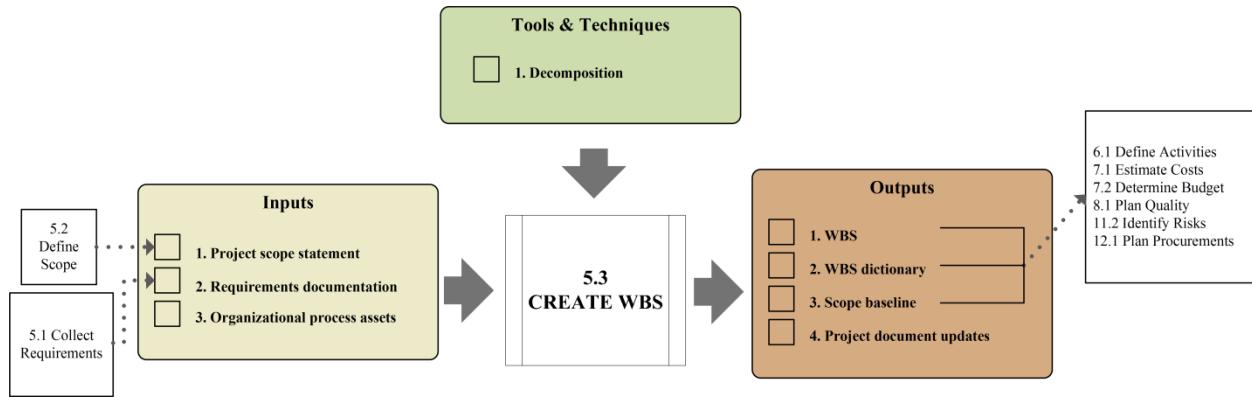
The WBS dictionary is a companion document to the work breakdown structure and they nearly always go hand-in-hand. A dictionary keeps the WBS from becoming visually cluttered, and it provides a document for the details and attributes about the WBS elements. Dictionary entries and the WBS elements are cross-referenced through a common numbering scheme. In practice, the WBS dictionary is normally begun at the same time as the work breakdown structure is being created.

Some of the information the WBS dictionary might contain about each component and work package can include:

- statement of work for the deliverable component
- expanded description of the work package, including important tasks or items that shouldn't be overlooked
- account codes, general ledger accounts, expense account numbers, labor accounts, or time entry accounts
- responsible parties or organization
- contact information

- quality requirements
- additional references to technical specifications or documents
- additional breakdowns of scheduled activities or costs.

### 5.3 Process decomposition



#### Inputs

##### Project scope statement

The project scope statement details the measurable goals, objectives, deliverables, and requirements of the project, and what the acceptance criteria of deliverables will be. It also describes the work required to meet all objectives and deliverables of the project, and it also contains milestones, assumptions, risks, and costs. The deliverables will be compared to the project scope statement to make sure they comply with it.

##### Requirements documentation

The product scope and project objectives are broken down into requirements and described in a collection of requirements documentation that's applicable to the project and requirement type. Requirements can be documented and described in a number of different levels (executive, summary, and detailed) and in a variety of methods (textually or visually).

##### Organizational process assets

Organizational process assets are the source of existing policies, processes, organizational data and knowledge. These assets include the entire collection of formal and informal methodologies, policies, procedures, plans, and guidelines, as well as the organization's "knowledge base," which includes historical performance data, labor information, service and maintenance history, issue and defect history, project files, and financial data. In developing requirements and the project scope, templates, checklists, and lessons learned from previous projects are useful.

### Tools and Techniques

**Decomposition**

Decomposition breaks down larger items into manageable components. For example, in the creation of the WBS the deliverables are decomposed into work packages. In the case of activity definition, the WBS work packages are decomposed into scheduled activities.

### Outputs

**Work breakdown structure**

The WBS is deliverables-oriented, graphical, hierarchical representation of the work required to fulfill the project scope statement.

**WBS dictionary**

The WBS dictionary is a companion, supporting document for the WBS, and it includes detailed information about the work packages, including account identifier, responsibilities, statements of work, and contact information.

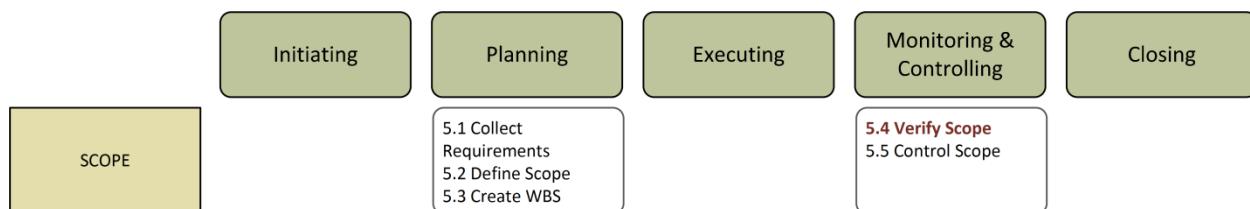
**Scope baseline**

The scope baseline is the approved project scope statement, WBS, and WBS dictionary. The baseline incorporates all approved changes.

**Project document updates**

Defining the WBS and WBS dictionary is likely to uncover additional requirements, so updates any project documents in existence would be needed.

### 5.4 Verify scope



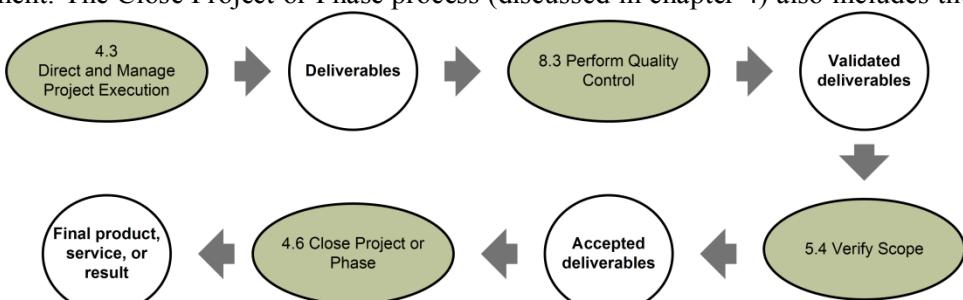
The **Verify Scope** process obtains the customer and stakeholders' acceptance of the completed deliverables. During scope verification the deliverables are compared to the scope statement to make sure all the requirements have been met. Scope verification is also performed by the project team throughout the project as deliverables are ready but before the stakeholders or customers verify the deliverables.

Since the project scope includes the requirements and acceptance criteria, it serves as the checkpoint. The deliverables are verified through inspection, which in this context means any tools that measure, examine, or verify that the work and deliverables meet the requirements and acceptance criteria. Though inspection may be more narrowly defined in some industries and application areas, its context here should be broadly interpreted.

Deliverables that don't meet these criteria result in change requests to address the deficiencies. Scope verification ultimately involves the stakeholders and customers, who are the ones who must agree that the deliverables meet their needs as established in the scope. But this process is also performed beforehand by the project team, usually after quality control. Scope verification always occurs at least once for each

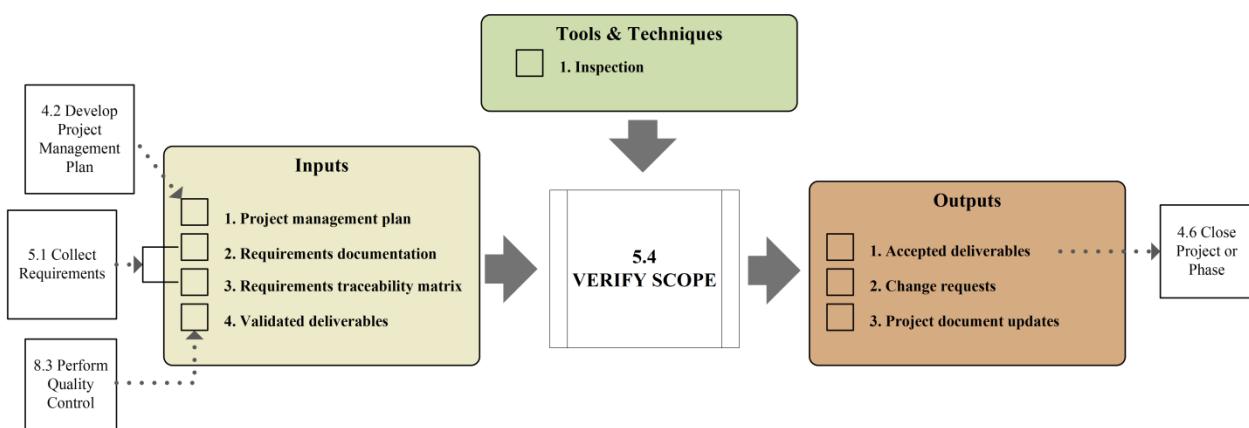
project, and even if the project is terminated before completion the deliverables are compared to the scope's requirements based on the work completed when the project was canceled.

It's probably best to think of scope verification as a "customer review" process because that helps eliminate some of the confusion with two other processes concerned with verification and user acceptance. Perform Quality Control (discussed in chapter 8) validates the deliverables against quality requirements while scope verification is concerned with the overall acceptance and verification against the project scope statement. The Close Project or Phase process (discussed in chapter 4) also includes the formal acceptance of the deliverables by the customer (through the Administrative Closure Procedure), but its focus is on the formalities required by the customer or the performing organization, such as a signed acceptance document.



PROCESS	PURPOSE	PRIMARY OUTPUT
Direct and Manage Project Execution	Creation of deliverables	Deliverables
Verify Scope	Acceptance by customer (adherence to scope)	Accepted deliverables
Perform Quality Control	Adherence to quality requirements	Validated deliverables
Close Project or Phase	Formal closure	Final product, service, or result

#### 5.4 Process decomposition



**Inputs** **Project management plan**

The project management plan details how the project will be executed, managed, and controlled. For scope verification, the project management plan contains the project scope statement, work breakdown structure, and the WBS dictionary. These are the scope baseline, and the validated deliverables will be compared to these documents for adherence to requirements.

 **Requirements documentation**

The product scope and project objectives are broken down into requirements and described in a collection of requirements documentation that's applicable to the project and requirement type. Requirements can be documented and described in a number of different levels (executive, summary, and detailed) and in a variety of methods (textually or visually).

 **Requirements traceability matrix**

All requirements are logged onto a requirements traceability matrix. The main purpose of the matrix is to enable every requirement to be logged and attached to a project objective or to another requirement in a hierarchical manner.

 **Validated deliverables**

These are deliverables that have met all the quality requirements through Perform Quality Control.

**Tools and Techniques** **Inspection**

Inspection is any activities that make sure the work and deliverables meet the requirements and the product acceptance criteria. Inspection usually involves measuring, examining, or other verification activities.

**Outputs** **Accepted deliverables**

Deliverables that have met project requirements and the product acceptance criteria are known as accepted deliverables.

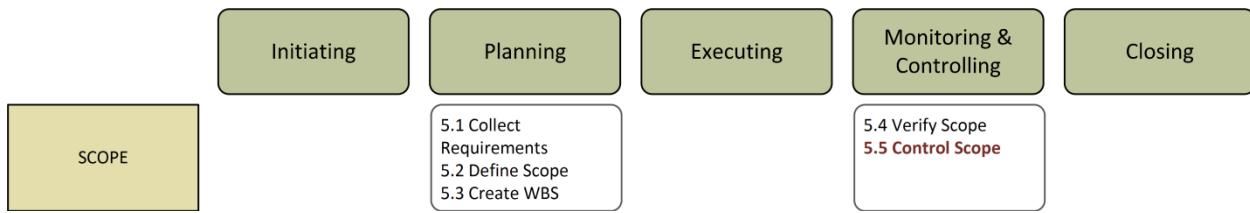
 **Change requests**

Deliverables that do not meet the project's requirements and the product acceptance criteria will result in change requests for correction and defect repair. These are treated as any other change request and processed through Perform Integrated Change Control.

 **Project document updates**

Project status information, change logs, and other project documents may need updated through scope verification activities.

## 5.5 Control scope

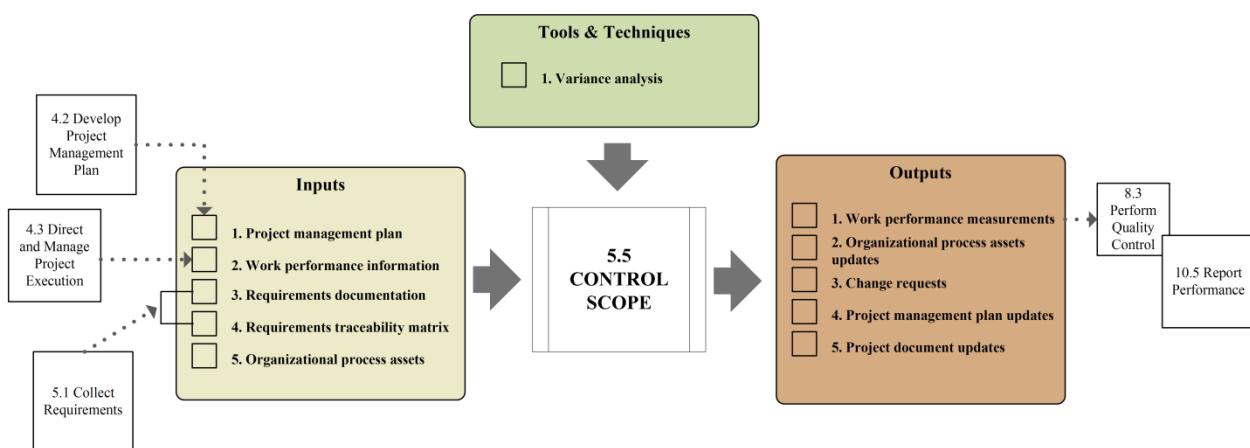


The **Control Scope process** is about two things: proactively influencing the factors that can contribute to scope changes and managing the impact of changes when they do occur. The project manager must do all within his or her abilities to keep changes from overwhelming the project, and that starts by having a solid project scope statement. Another way to help influence change is to be engaged throughout the project with the customer, stakeholders, and project team. By keeping an open dialogue, the project manager can often “sniff-out” probable scope changes before they’re requested. The customer isn’t the only source for requested scope changes: the sponsor, organizational management, stakeholders, and the project team often request expansions to the project scope.

Project scope control makes sure that scope change requests are coordinated through the change control system. Approved scope changes have to be managed rigorously, communicated to the stakeholders and project team, and incorporated into all effected components of the project management plan. Approved scope changes will result in updates to the overall project performance baseline.

Scope control also monitors performance to look for scope variances between what is occurring to what was planned for. This is done by comparing the scope baseline to the work being completed. When variances are found, the reasons need to be understood and documented, and they may result in change requests for preventative or corrective actions to bring actual performance back in line with what was planned, or to alter the plan to reflect a changed situation.

## 5.5 Process decomposition



**Inputs** **Project management plan**

The project management plan details how the project will be executed, managed, and controlled. For scope control, several subsidiary elements of the plan will be referred to, including the scope baseline, scope management plan, change management plan, configuration management plan, and the requirements management plan.

 **Work performance information**

Work performance information is any data that can be considered related to the work which produces the project deliverables. Examples are schedule and progress status information, budget and cost status, quality status, estimates to complete, resource utilization information, and lessons learned. This information helps identify variances.

 **Requirements documentation**

The product scope and project objectives are broken down into requirements and described in a collection of requirements documentation that's applicable to the project and requirement type. This information is needed to help identify variances.

 **Requirements traceability matrix**

All requirements are logged onto a requirements traceability matrix. The main purpose of the matrix is to enable every requirement to be logged and attached to a project objective or to another requirement in a hierarchical manner.

 **Organizational process assets**

Organizational process assets are the source of existing policies, processes, organizational data and knowledge. The organization may have formal or informal procedures for reporting methods or scope control policies, procedures, or guidelines.

**Tools and Techniques** **Variance analysis**

Variance analysis uses appropriate methods to measure the differences between what was planned for and what is actually occurring. The root cause of variances will also need identified. Variances can result in a need for preventative or corrective actions, or scope changes to bring the plan back in line with what is happening.

**Outputs** **Work performance measurements**

This includes work performance information that specifically provides mathematical measurements of performance that is communicated to stakeholders.

 **Organizational process assets updates**

Scope control adds to the organizational body of knowledge through variance analysis and lessons learned.

 **Change requests**

Scope control variances can result in requests for corrective or preventative actions, or requests to change the project management plan.

**Project management plan updates**

Approved change requests will result in updates to the scope baseline and may also necessitate updates to other project performance baselines.

**Project document updates**

Scope control can result in updates to other project documents, including the requirements documentation.

### References

1. Bauer, Martin. (2005). Requirements Gathering Essentials. Retrieved from <http://www.sitepoint.com/article/requirements-gathering>.
2. McGannon, Bob, PMP. (n.d.). Avoiding the Dark Twisty Turn-Filled Tunnel Syndrome. Retrieved from <http://www.projectsmart.co.uk/avoiding-the-dark-twisty-turn-filled-tunnel-syndrome.html>.
3. Stephens, Matt. (2007). Grappling With Eels - First Wriggle: The Slippery World of Agile Planning. Retrieved from [http://www.regdeveloper.co.uk/2007/05/10/grappling\\_eels\\_one/](http://www.regdeveloper.co.uk/2007/05/10/grappling_eels_one/).
4. de Bono, Edward. (n.d.). Passes of the Day. Retrieved from [http://www.edwarddebono.com/PassageDetail.php?passage\\_id=963&](http://www.edwarddebono.com/PassageDetail.php?passage_id=963&).
5. Black, Ron. (2004). *The Complete Idiot's Guide to Project Management with Microsoft Project 2003*. Page 83. New York: Alpha Group.
6. Karlsson, J. "Towards a Strategy for Software Requirements Selection. Licentiate." Thesis 513, Linköping University, October 1995.
7. Leffingwell, D. & Widrig, D., *Managing Software Requirements: A Use Case Approach*, 2nd ed. Boston, MA: Addison-Wesley, 2003.

**Chapter summary**

Project scope management ensures that only the necessary work is performed in order to fulfill the project's requirements. In order to determine exactly what deliverables are needed and the work that must be performed, requirements must be known. Preplanning for requirements is done through the creation of the requirements management plan. Requirements are collected, documented, and prioritized through the Collect Requirements process, which results in requirements documentation and the requirements traceability matrix.

The scope management plan lays out how the project scope, work breakdown structure, and WBS dictionary will be created. It also defines how the scope will be managed and controlled, and how changes will be addressed through the scope change control system. Templates are useful for getting the scope management plan started.

The project scope statement describes the work required to meet the project's objectives and requirements. It is created in the Define Scope process and uses the requirements, product analysis, and alternatives identification. The project scope statement contains a significant amount of information. At a minimum it includes the product scope, product acceptance criteria, a description of all project deliverables, any items excluded from the scope, project constraints, and project assumptions.

Once the scope statement has been initially created, the work breakdown structure is created. The WBS is a, deliverables-oriented, hierarchical view of the deliverables and requirements. It consists of components, work

packages, planning components, and planning packages. Work packages are the lowest level of work within the WBS that can be easily managed. The WBS shouldn't be confused with an activity or task list. Templates or work breakdown structures from similar projects can help get its creation started.

The WBS dictionary is a companion document to the WBS. The WBS is cross-referenced to the components and work packages on the WBS through a numbering system. The WBS dictionary contains additional information about each WBS element. This could be expense accounts, general ledger accounts, time accounts, additional specifications, contact information, and responsible individuals.

The deliverables are checked for adherence to requirements through the Verify Scope process. The deliverables are verified against the scope using inspection. Deliverables that fail validation can result in recommended corrective actions. The scope verification process can be thought of as the customer review process.

Activities in the Control Scope process serve two functions: the first is to proactively influence the factors that can contribute to scope changes, and the second is managing the impact of the changes when they occur by ensuring that scope changes are funneled through the scope change control system (which is a component of the change control system). Scope control also uses variance analysis to compare what's actually occurring to what was planned for. Variances can necessitate the need for requested changes.

**Exam summary**

- Project scope management is about making sure the scope is thoroughly defined and that unnecessary changes to it are prevented.
- The requirements management plan describes the processes and procedures for how requirements will be collected, categorized, and prioritized.
- The requirements management plan is a subsidiary component of the project management plan.
- The scope management plan is the planning document for how the scope will be defined, monitored, and controlled. It also describes the process for formal acceptance of the product.
- The scope management plan is a subsidiary component of the project management plan.
- The scope change control system identifies the process for how scope change requests will be submitted, logged, documented, reviewed, and approved or denied. It is part of the overall project change control system and configuration management system.
- The scope change control system is part of the project's change control system.
  
- The project scope is the work required, and only the work required, for the project to meet its objectives.
- Product analysis includes the methods for deconstructing the product and thoroughly understanding it.
- Alternatives identification includes methods for finding different solutions to a proposed solution.
- Product analysis and alternatives identification are tools that help identify requirements for the project scope statement.
- The project scope statement is created by the Define Scope. At a minimum the project scope contains:
  - Product scope
  - Product acceptance criteria
  - Project deliverables
  - Project exclusions
  - Project constraints
  - Project assumptions
  
- The work breakdown structure is a deliverables-oriented, visual presentation of project components.
- The smallest level of manageable work within the WBS is the work package.
- The primary input to the Create WBS process is the project scope statement.

- The WBS is an input to activity definition, cost estimating, cost budgeting, resource planning, and risk management planning.
- Decomposition takes deliverables and work and breaks it down into successively smaller chunks until it can be managed, scheduled, assigned, and estimated.
- The WBS dictionary is a companion document to the WBS. It numerically references each WBS component and work package, and it contains additional information about each WBS element.
- The scope baseline is the collection of the project scope statement, the work breakdown structure, and the WBS dictionary.
- The scope baseline is created when the project scope statement is approved.
- The scope baseline is a component of the project performance baseline.
- Baselines are the current, approved plans, which include approved changes.
- Through inspection, the scope verification process evaluates the deliverables to the project scope statement.
- For exam purposes, consider that if the project scope is completed and has been verified that the project is complete.
- The Control Scope process is about preventing unnecessary changes, and making sure that approved changes are integrated throughout the project management plan.
- Scope control begins immediately when the scope baseline has been created (the project scope statement has been approved).
- Scope control uses the project scope change control system, change control system, and configuration management system.
- Within scope control, variance analysis looks at what is currently occurring and compares it to what was planned.
- Variance analysis can result in change requests.