



Chapter Five-Scope management

Eskinder Mesfin Cherinet(PMP|PPRINCE2|MCSE|CCNP|ITIL)

Chapter Five-Scope management

Confused about exactly what you should be working on?

- Once you have a good idea of what needs to be done, you need to **track your scope as** the project work is happening. As each goal is accomplished, you confirm that all of the work has been done and make sure that the people who asked for it are **satisfied with the result.**
- **In this chapter, you'll learn the tools that help your project team set its goals** and keep everybody on track.

You've got to know what (and how) you
will build before you build it

It looks like we have a scope problem

The product scope is all about the final product—its features, components, pieces.

When people talk about scoping out their products, a lot of times they're talking about figuring out the features of the product, not the work that goes into it.

Product scope means the features and functions of the product or service that you and your team are building.

When we talk about scoping out a project, we mean figuring out all of the work that needs to be done to make the product.

Project scope is all of the work that needs to be done to make the product.

THIS is a big part of what the project manager is concerned with...the work the team has to do.

Scope creep means uncontrolled changes that cause the team to do extra work.

This means changes that just went in without anyone bothering to figure out what effect they'd have on the project's time, cost, scope, quality, risk, or resources.

Try?

1. Programming

☐ Project scope

☐ Product scope

2. 34 levels in the game

☐ Project scope

☐ Product scope

3. Graphic design

☐ Project scope

☐ Product scope

4. Four playable characters

☐ Project scope

☐ Product scope

5. Great graphics

☐ Project scope

☐ Product scope

6. Testing

☐ Project scope

☐ Product scope

7. Mac and PC compatible

☐ Project scope

☐ Product scope

8. A “boss battle” milk fight level at the end

☐ Project scope

☐ Product scope

You've got to know what (and how) you will build before you build it

- You always want to know exactly what work has to be done to finish your project *before you start it*. You've got a bunch of team members, and you need to know exactly what **they're going to do to build your product**. So how do you write down the scope?
- That's the goal of the **six Scope Management processes**. They're **about** figuring out how you will identify all of the **work your team will do during the project**, coming up with a way to make sure that you've written down what work will be done (and nothing else!), and making sure that **when things change on your project**, you keep its scope up to date so that your team is always building **the right product**

The Scope Management plan describes how you write down the scope, make sure it's right, and keep it up to date.

The power of Scope Management

- When you take control of your project's scope, you're doing more than just planning. It turns out that when projects have scope problems, the results are actually pretty predictable. Take a look at these problems that your project team ran into. Do any of these sound familiar to you? Many project managers run into similar problems on their own projects.

Scope Management(Probl

- 1 **The team had trouble getting the project off the ground.** Everyone on the team was good at their individual jobs, but it seemed like nobody knew how to get the project started.

They'd sit around in meetings talking about what they wanted to build, but it seemed like weeks before anything started getting done.

- 2 **There were a lot of false starts.** Just when they thought they were getting the project under way, it seemed like something would shift and they'd be back to square one.

- 3 **The sponsor and stakeholders were unpredictable.** There were three different times that Amy and Brian thought they were done. But each time, a stakeholder found a problem that sent them back to the drawing board.

The worst part about this was that there was no way to know when they were done with the project without asking for the sponsor's opinion...and it seemed like that opinion was always changing.

- 4 **There were a whole lot of changes.** They were always scrambling to keep up with shifting priorities and ideas, and they never knew for sure what they'd be working on each week.

The team was tempted to lay down the law and forbid any changes...but a lot of those changes were necessary, and good ideas.

The six Scope Management processes



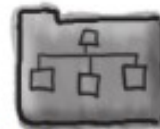
Project Management plan



Requirements documentation



Project scope statement



Work breakdown structure



Change requests



Accepted deliverables

Plan Scope Management

Here's where you write down the subsidiary plan for the project management plan that we talked about in the last chapter. You plan out all of the work you'll do to define your scope, make sure the team is planning to do the right work, and control it.

Collect Requirements

In this process, you find out all of the stakeholders' needs and write them down so that you know what to build and your requirements can be measured and tracked.

Define Scope

Here's where you write down a detailed description of the work you'll do and what you'll produce.

When you do this right, the stakeholders are never unpredictable because you already understand their needs.

Create WBS

The work breakdown structure (WBS) organizes all of your team's work into work packages—or discrete pieces of work that team members do—so that you can keep the momentum of the project going from the start.

Control Scope

We already know how important it is to control changes on your project. When scope changes aren't controlled, it leads to the most frustrating sort of project problems. Luckily, you already know about change control, and now you can use it to manage your project's scope.

Pay attention to the WBS—there will be a lot of questions about it on the exam.

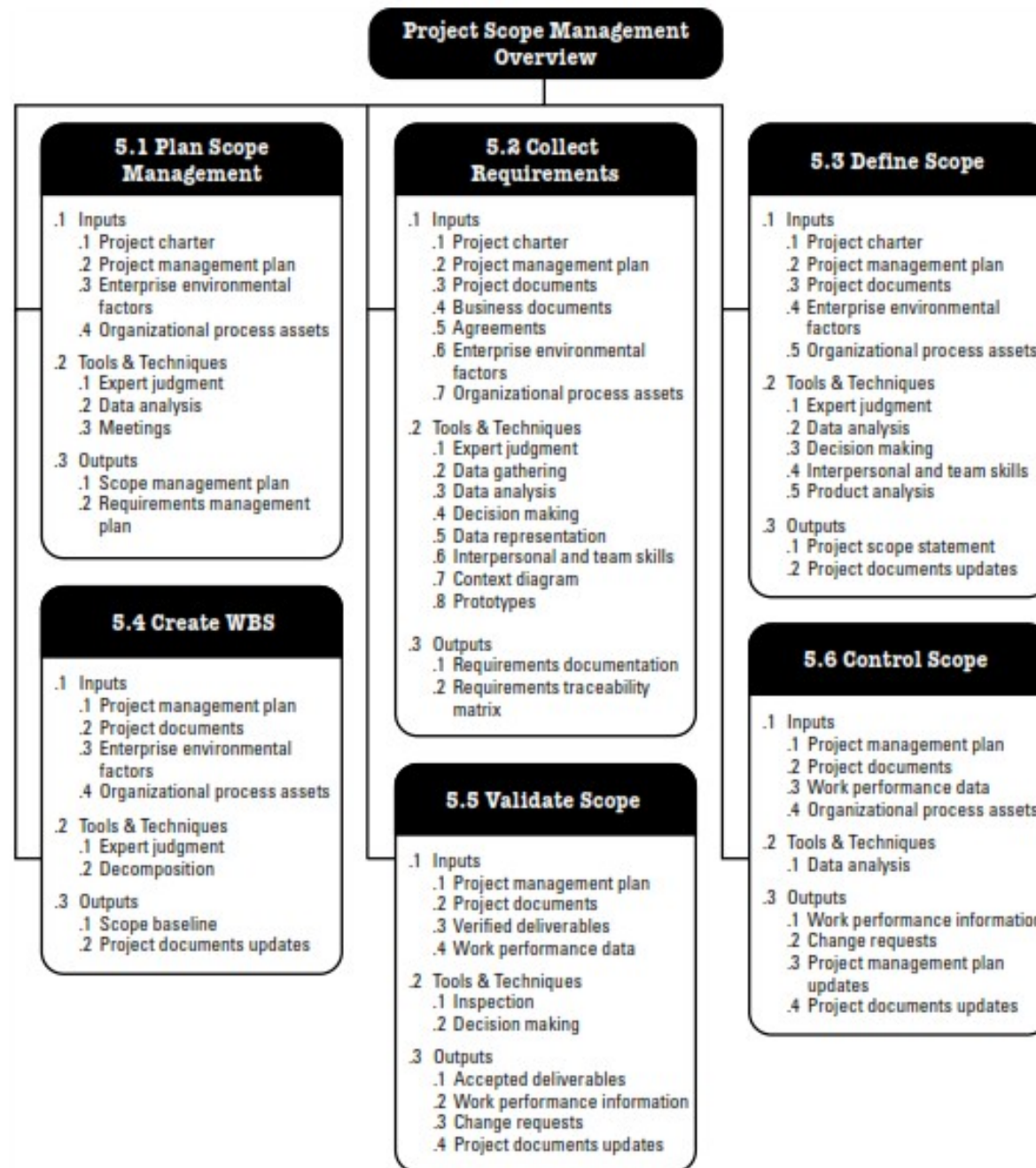
Validate Scope

Once the work is complete, you need to make sure that what you're delivering matches what you wrote down in the project scope statement. That way, the team never delivers the wrong product to the customer.

Remember integrated change control from Chapter 4? Now you'll see it in action.

On the exam, "customer" can mean the same thing as "client" and "sponsor."

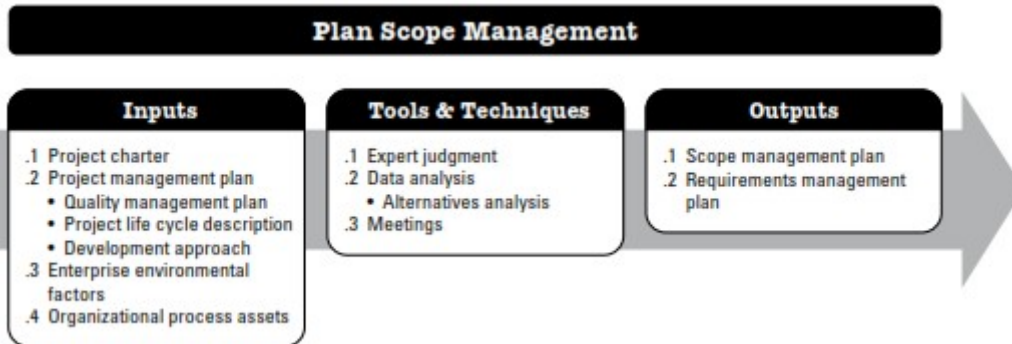
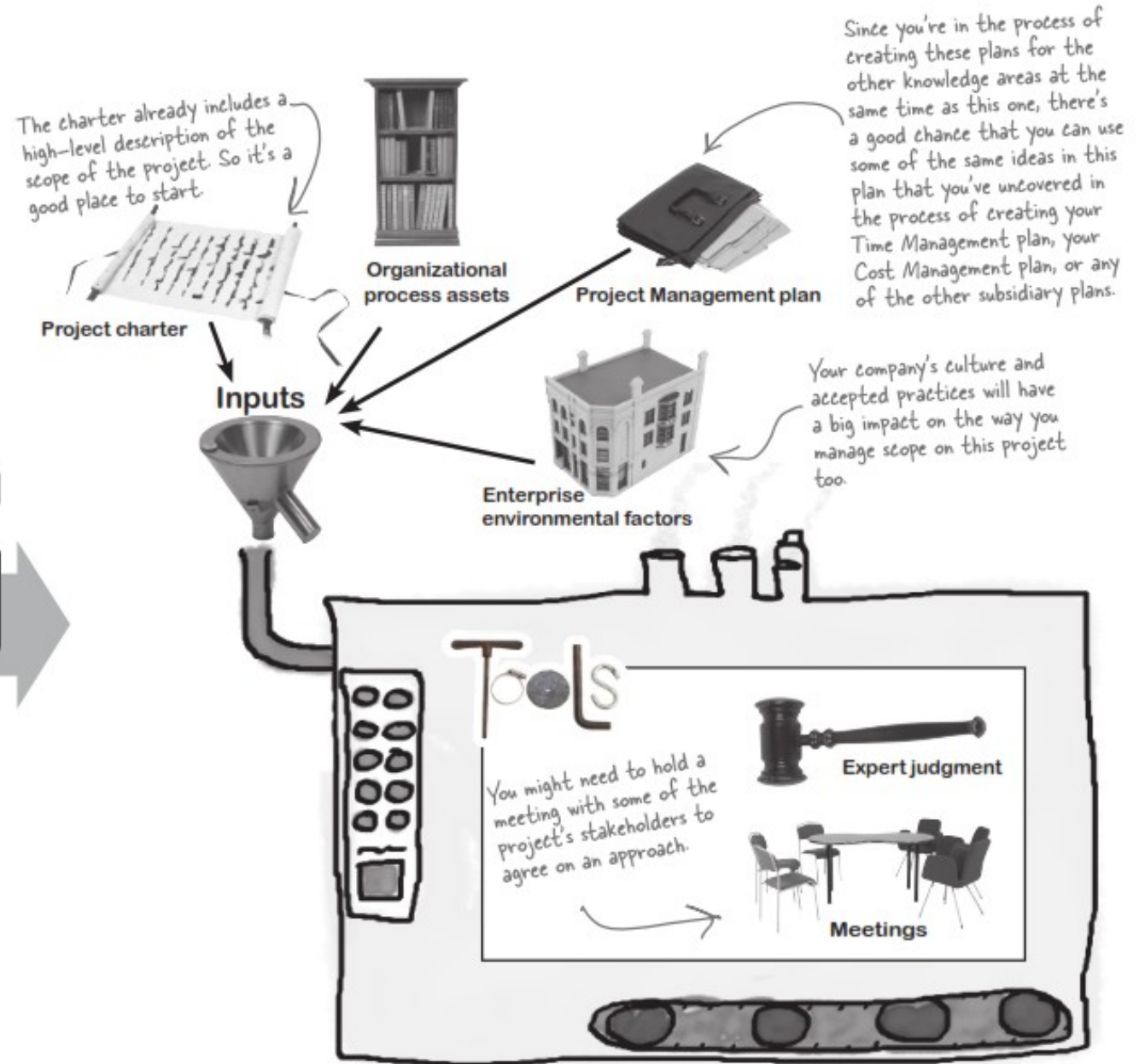
Overview



Plan your scoping processes

- Here's where you figure out how you'll approach defining and validating the scope of your project. The **Plan Scope Management process is where you lay out your** approach to figuring out what work you'll do and what's out of scope.
- All of the other processes in the Scope Management knowledge area are defined and described in this document.
- It's the blueprint you'll use for everything else you'll do to manage scope through the project.

Plan scope management



Plan scope Management Output

There are two outputs of the Plan Scope Management process: the **Scope Management plan** and the **Requirements Management plan**.

Both of them help you define the scope of your project and make sure that you and your team are focused on only the work that will help you satisfy your customers' needs.



Your Requirements Management plan will describe all of the processes your team will use to document your requirements and maintain that document throughout the project.

Requirements Management plan

Here's where you'll find a description of the approach the team will take to planning, tracking, and reporting on requirements. You'll use this document to describe the prioritization process for requirements, and how you'll build a traceability matrix for your requirements as well.



The Scope Management plan isn't just about writing a scope document; it details the process you use to come up with your work breakdown structure too.

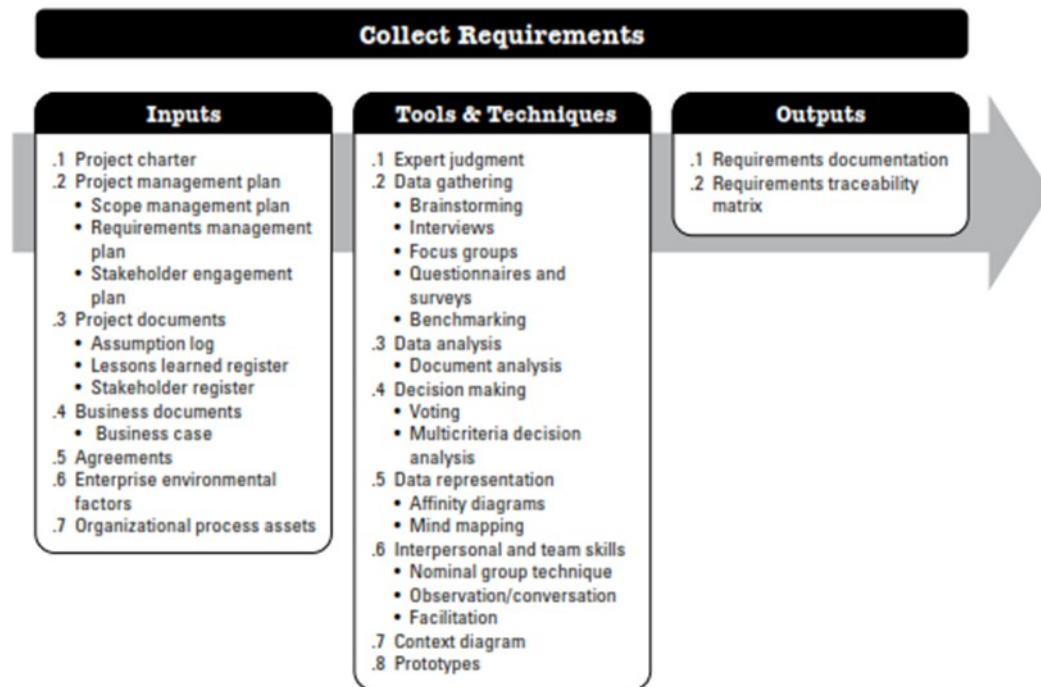
Scope Management plan

Here's where you write down the subsidiary plan for the Project Management plan that we talked about in Chapter 4. You plan out all of the work you'll do to define your scope, with the right work planned for the team, and control it.

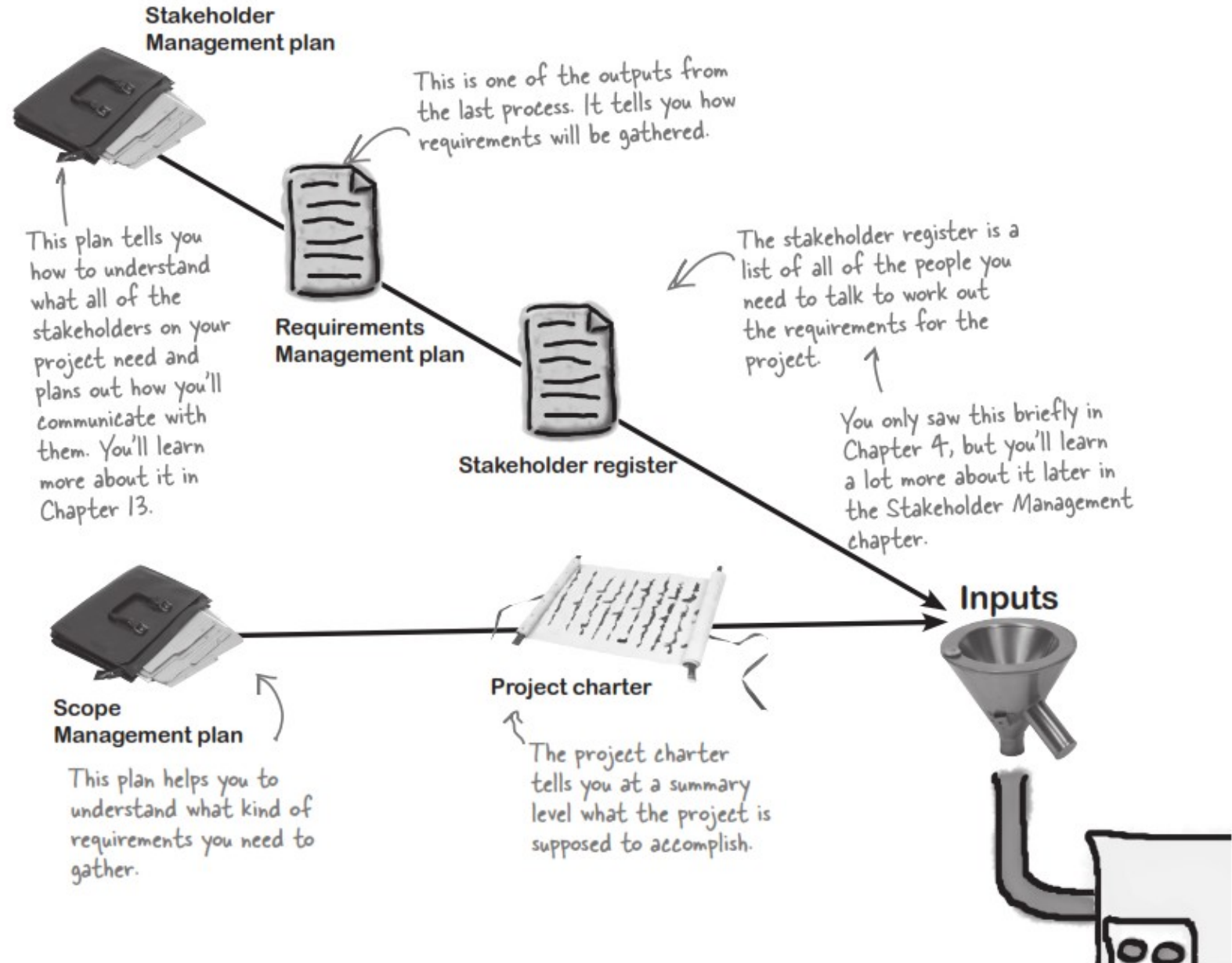
The Plan Scope Management process helps you think through everything you'll need to do to keep your project focused on the right work from beginning to end.

Collect requirements for your project

- Gathering requirements is all about sitting down with all of the stakeholders for your project and working out what their needs are, and that's what you do in the **Collect Requirements process**.
- **If your project is going to be** successful, you need to know what it will take for all of your stakeholders to agree that your project has met its goals. You need to have a good idea of what's required of your project up front, or you'll have a tough time knowing whether or not you're doing a good job as you go. That's why you need to write down all of your project and product requirements with enough detail that you can measure your team's progress.



Collect requirements Process



Talk to your stakeholders

- The **Collect Requirements process involves talking to the people who are affected by your** project to find out what they need. All of the tools in this process are focused on getting your stakeholders to tell you about the problem that the project is going to solve.
- Sometimes that means sitting down with each of them one-on-one, and other times you can do it in a group setting. One of the most important things to understand about requirements is that every requirement fulfills a specific stakeholder need.
- But that's not the only place you'll find requirements, so here are three really useful tools and techniques to help you gather requirements:

Talk to your stakeholders

Interviews are important ways to get your stakeholders to explain how they'll use the product or service your project is creating. By talking to people one-on-one, you can get them to explain exactly what they need so that you can be sure that your project can meet its goals.

Focus groups are another way to get a group of people to discuss their needs with you. By letting a group discuss the end product together, you can get them to tell you requirements that they might not have thought of by themselves.

Facilitated workshops are more structured group conversations where a moderator leads the group through brainstorming requirements together. In facilitated workshops, misunderstandings and issues can get reconciled all at once because all of the stakeholders are working together to define the requirements.

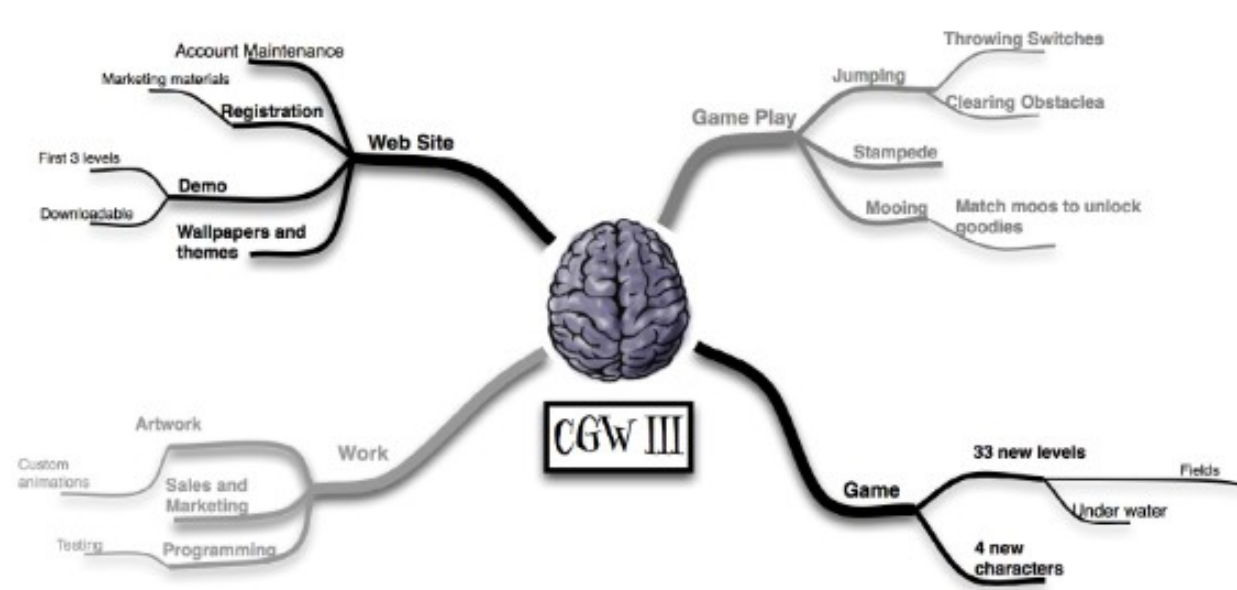
Make decisions about requirements

- A big project usually has a lot of stakeholders, and that means a lot of opinions. You'll need to find a way of making decisions when those opinions conflict with each other. There are four major decision-making techniques you can choose from. These are referred to as **group decision-making techniques on the test**.
 - **Unanimity** means everyone agrees on the decision.
 - **Majority** means that more than half the people in the group agree on the decision.
 - **Plurality** means that the idea that gets the most votes wins.
 - **Dictatorship** is when one person makes the decision for the whole group.

Help your team to get creative

- Getting your team to think creatively can help you create a better product from the start. **Group creativity techniques are all about getting those** creative juices flowing while you gather your requirements.

Idea/mind maps are a good way to visualize the way your ideas relate to each other. When you've finished working through an idea, it sometimes helps to create a map of how you got there and show which ideas can be grouped together.



Help your team to get creative

The Delphi technique is a way of letting everyone in the group give their thoughts about what should be in the product while keeping them anonymous. When you use the Delphi technique, everybody writes down their answers to the same questions about what the product needs to do and then hands them into a moderator. The questions could be about specific features that the product should have.

When the CGW team used the Delphi technique, here were a few of their questions:

CGW III Delphi Questionnaire

- What new levels would you like to see in the game?
- What new abilities should Bessie have?
- What should the story for *CGW III* be about?

The Delphi technique can be used to estimate the work the team will need to do and how long it will take too!



The moderator keeps everybody's names to himself, but shares the ideas so that everyone can learn from them and think of new ones. After everybody discusses those ideas, they're given a chance to adjust their original answers to the questions and hand them back in to the moderator. These iterations continue a few times until the group settles on a list of requirements for the product.

Help your team to get creative

Affinity diagrams are great when you have a lot of ideas and you need to group them so you can do something with them. A lot of people make affinity diagrams using Post-it notes on walls. That way, you can move the ideas around and change the groupings when you think of new areas to explore. Sometimes just putting requirements in categories will help you to find new ones.

The nominal group technique is a form of brainstorming where you write down the ideas as you find them and have the group vote on which ones they like the best. You then use the votes to rank all of the ideas and separate the ones that aren't important from the ones you want to delve into deeper.

Context diagrams help your team show the way all of the processes and features in your product scope relate to each other. It's a picture of the scope of your product that shows how users will interact with it.

Brainstorming is one of the most commonly used ways of collecting requirements. Whenever you sit a group of people down to think of new ideas, you're brainstorming.

Benchmarking is a way of comparing the processes and practices used in building your software with the practices and processes in other organizations so you can figure out the best ideas for improvement.

Document analysis is a way of collecting requirements by reading through all of the existing documents for your product.

A prototype shows users what your product will be like

- Sometimes the best way to get your stakeholders to give you an opinion on what your product should be is to show it to them in a **prototype**.
- **Prototypes are models of the** product that you're going to build that give your stakeholders a better idea of what your team is thinking.
- Sometimes users who are experimenting with a prototype will come up with a brand new requirement that they never thought of before. If you can get them to find it in the prototype, it's a lot easier to deal with than if you wait until the end of the project to show to them. When you're making a really complicated product, it can make sense to prototype it as part of the requirements collection process so that you can find changes that your users will ask for early on.

Output

Functional requirements are most of the kinds of things that you think of right away: new features, bug fixes, and new or different behavior.

Nonfunctional requirements are sometimes called quality attributes because they're things that you expect from your deliverables, but aren't specific features. Some examples of nonfunctional requirements are performance, reliability, error handling, and ease of use.

Outputs



This requirement is measurable. If the end product has puzzles that involve swimming, the requirement will pass its test. If not, it will fail.

Here, you can load the levels and time it to figure out if the product meets its requirements.

CGW III Requirements Document

1. Introduction
CGW II was a huge hit. We've done some market research and some internal brainstorming and compiled these requirements for *Cows Gone Wild III: The Milkening*, which will be released next year in time for the holidays.

2. Organizational Impact
This product will have an impact on many departments at Ranch Hand Games, including Research and Development, Marketing, Distribution, Shipping, Administration, Finance, and Customer Service.

3. Functional requirements

Name	RU001—Include Underwater levels.
Summary	The cows will need to be able to move around under water.
Rationale	Underwater environment was the single biggest request from polled gamers.
Requirement	Cows will need to be able to swim, and underwater puzzles will need to be developed that require swimming.

...

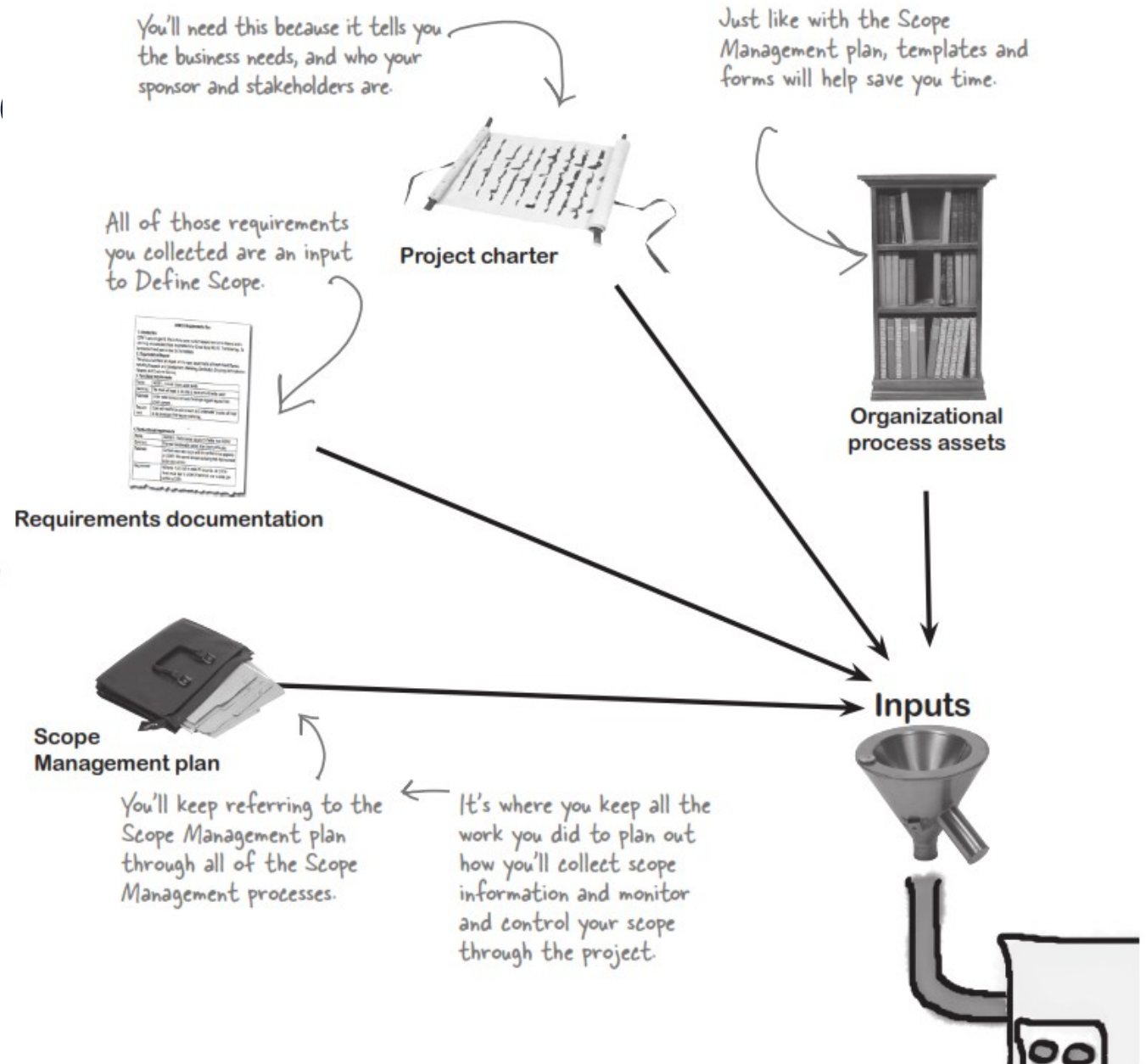
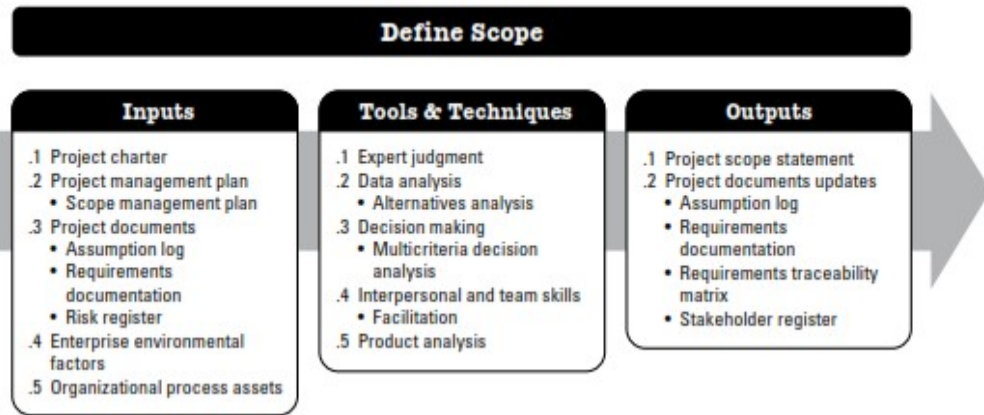
4. Nonfunctional requirements

Name	RNF001—Performance as good or better than CGW II.
Summary	The new functionality cannot slow down game play.
Rationale	Gamers were very happy with the performance upgrades in CGW II. We cannot be seen as losing that improvement in the next version.
Requirement	All levels must load in under 15 seconds. All online levels must load in under 25 seconds over a cable connection at 256K.

Requirement Traceability Matrix

Requirements Traceability Matrix								
Project Name:								
Cost Center:								
Project Description:								
ID	Associate ID	Requirements Description	Business Needs, Opportunities, Goals, Objectives	Project Objectives	WBS Deliverables	Product Design	Product Development	Test Cases
001	1.0							
	1.1							
	1.2							
	1.2.1							
002	2.0							
	2.1							
	2.1.1							
003	3.0							
	3.1							
	3.2							
004	4.0							
005	5.0							

Define the scope of the project



How do you define the scope?

- You already got a head start on defining the project scope when you wrote down the requirements.
- But now you need to go a lot further and write down all of the work that you and your team are going to do over the course of the project.
- Luckily, the **Define Scope process tools and techniques are there to help guide you through creating the project** scope statement.
 - **Facilitated workshops**
 - **Product analysis**
 - **Alternatives generation**
 - **Expert judgment**

Outputs

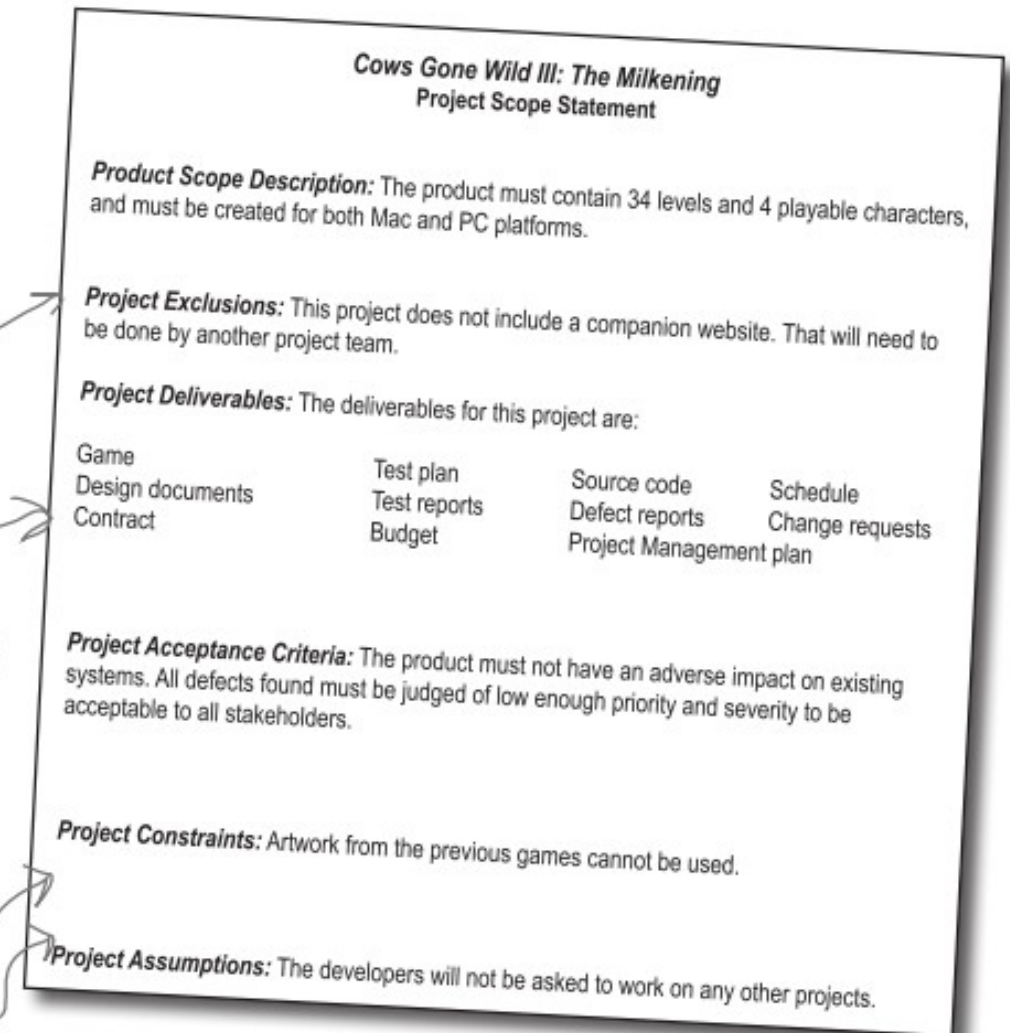
The project scope statement tells what work you are—and are not—going to do in the project.



This means looking for all the work the project **DOESN'T** include.

The deliverables listed here are **EVERYTHING** the project creates, including project management stuff.

Constraints are known limitations. Assumptions are things you think are true.



This is the other output of the process, and it has to do with change control. We'll get to that when we talk about the Control Scope process.



Project document updates

Try?

1. The work required to create the graphics

☐ Requirements document ☐ Project scope statement

2. New characters in the game

☐ Requirements document ☐ Project scope statement

3. 33 new levels

☐ Requirements document ☐ Project scope statement

4. The performance requirements for the product

☐ Requirements document ☐ Project scope statement

5. A description of how the WBS is created

☐ Requirements document ☐ Project scope statement

6. How the software will be tested

☐ Requirements document ☐ Project scope statement

7. How the stakeholders will verify the deliverables

☐ Requirements document ☐ Project scope statement

8. A list of all artwork that will be created

☐ Requirements document ☐ Project scope statement

Answers: Requirements doc: 2, 3, 4, 8 Project scope statement: 1, 5, 6, 7

Charter vs scope Statement

Project Charter

Project purpose

Measurable project objectives and related success criteria

High-level requirements

High-level project description, boundaries, and key deliverables

Overall project risk

Summary milestone schedule

Preapproved financial resources

Key stakeholder list

Project approval requirements (i.e., what constitutes success, who decides the project is successful, who signs off on the project)

Project exit criteria (i.e., what are the conditions to be met in order to close or to cancel the project or phase)

Assigned project manager, responsibility, and authority level

Name and authority of the sponsor or other person(s) authorizing the project charter

Project Scope Statement

Project scope description (progressively elaborated)

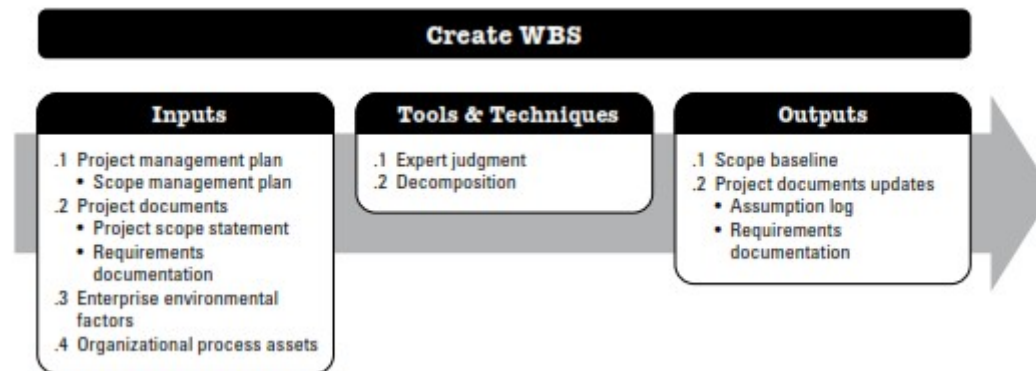
Project deliverables

Acceptance criteria

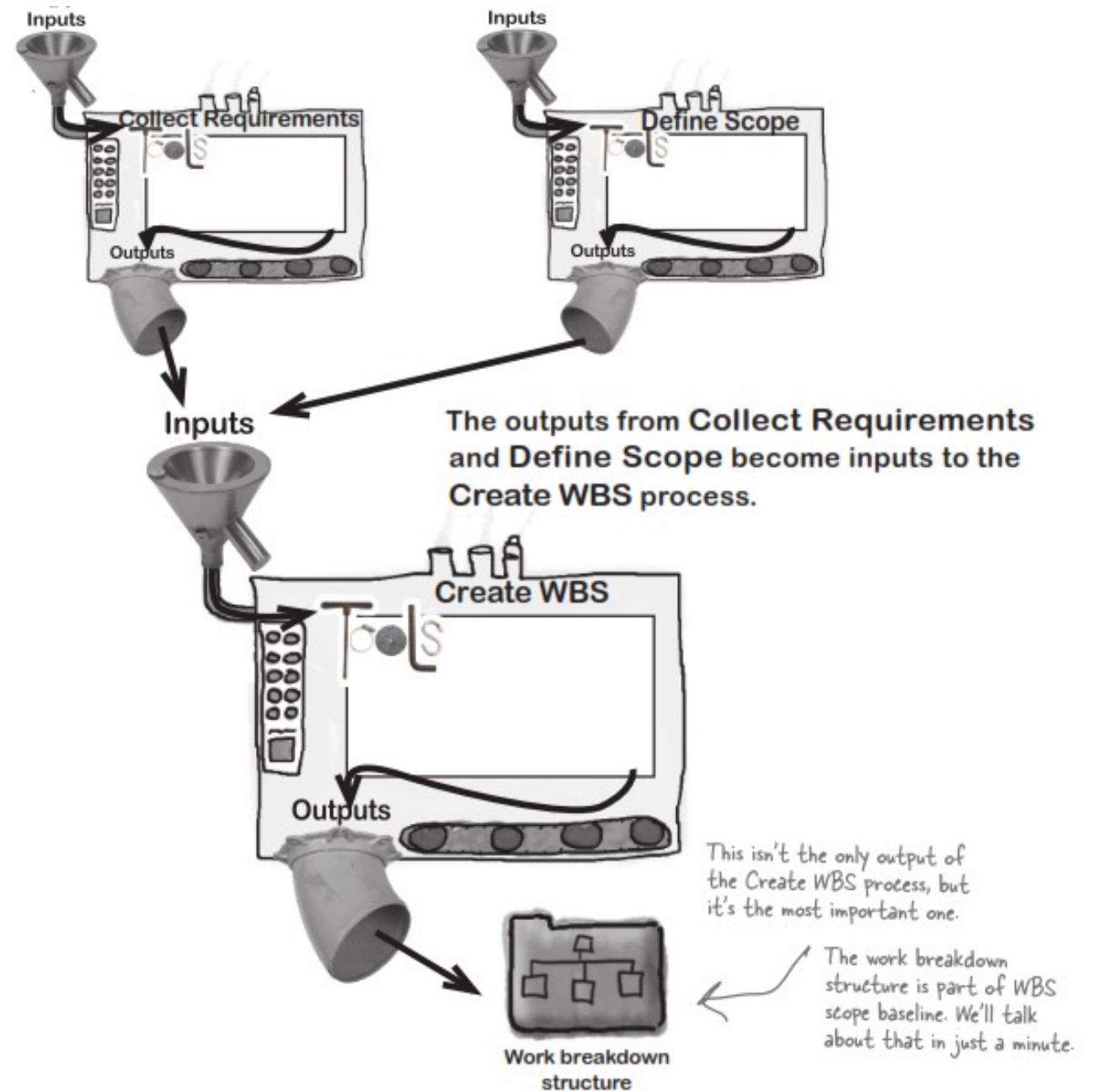
Project exclusions

Create the work breakdown structure

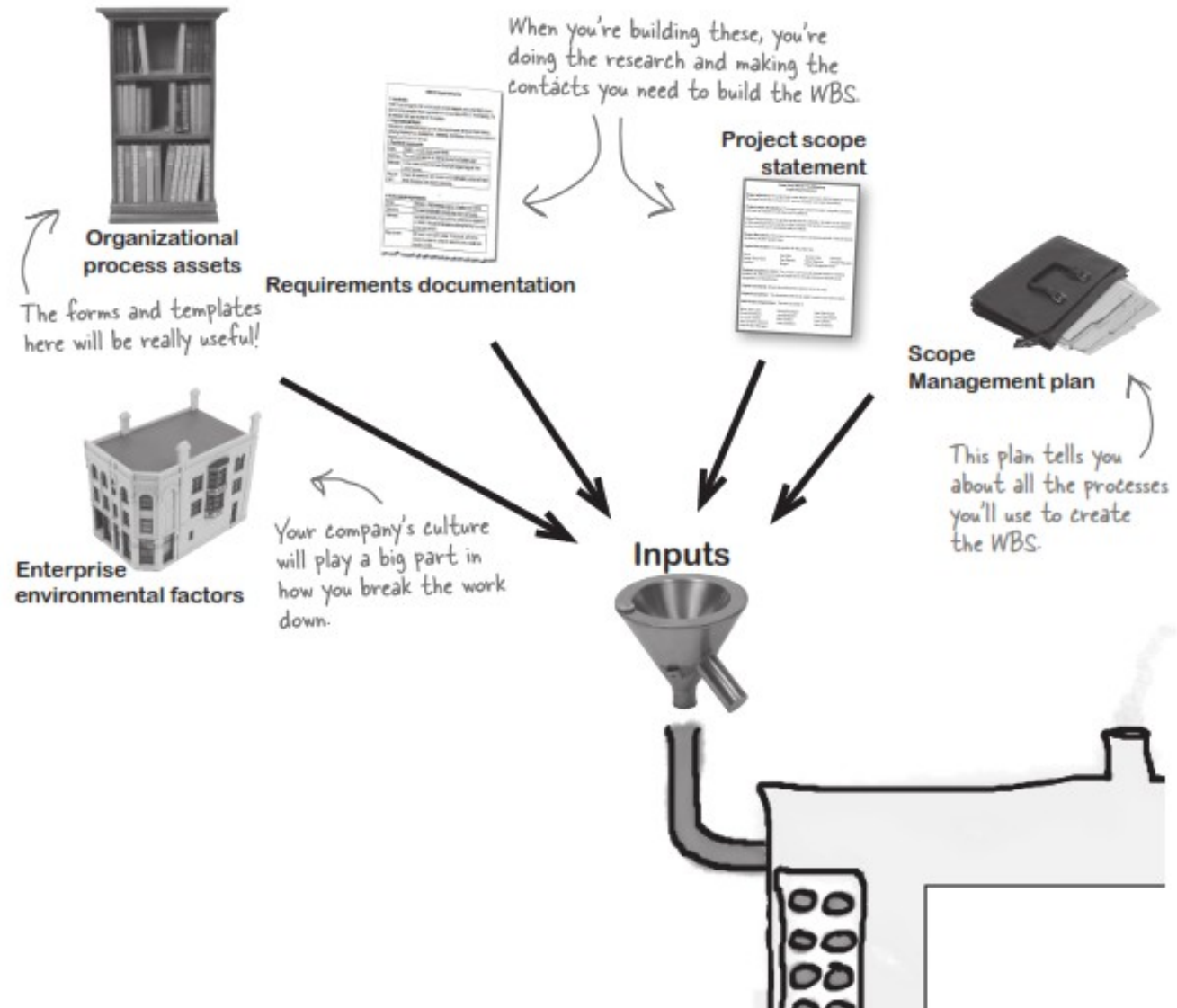
- The **Create WBS process is the most important process in the Scope Management** knowledge area because it's where you actually figure out all the work you're going to do. It's where you create the **work breakdown structure (or WBS), which is the** main Scope Management output. Every single thing that anyone on the project team—including you—will do is written down in the WBS somewhere.



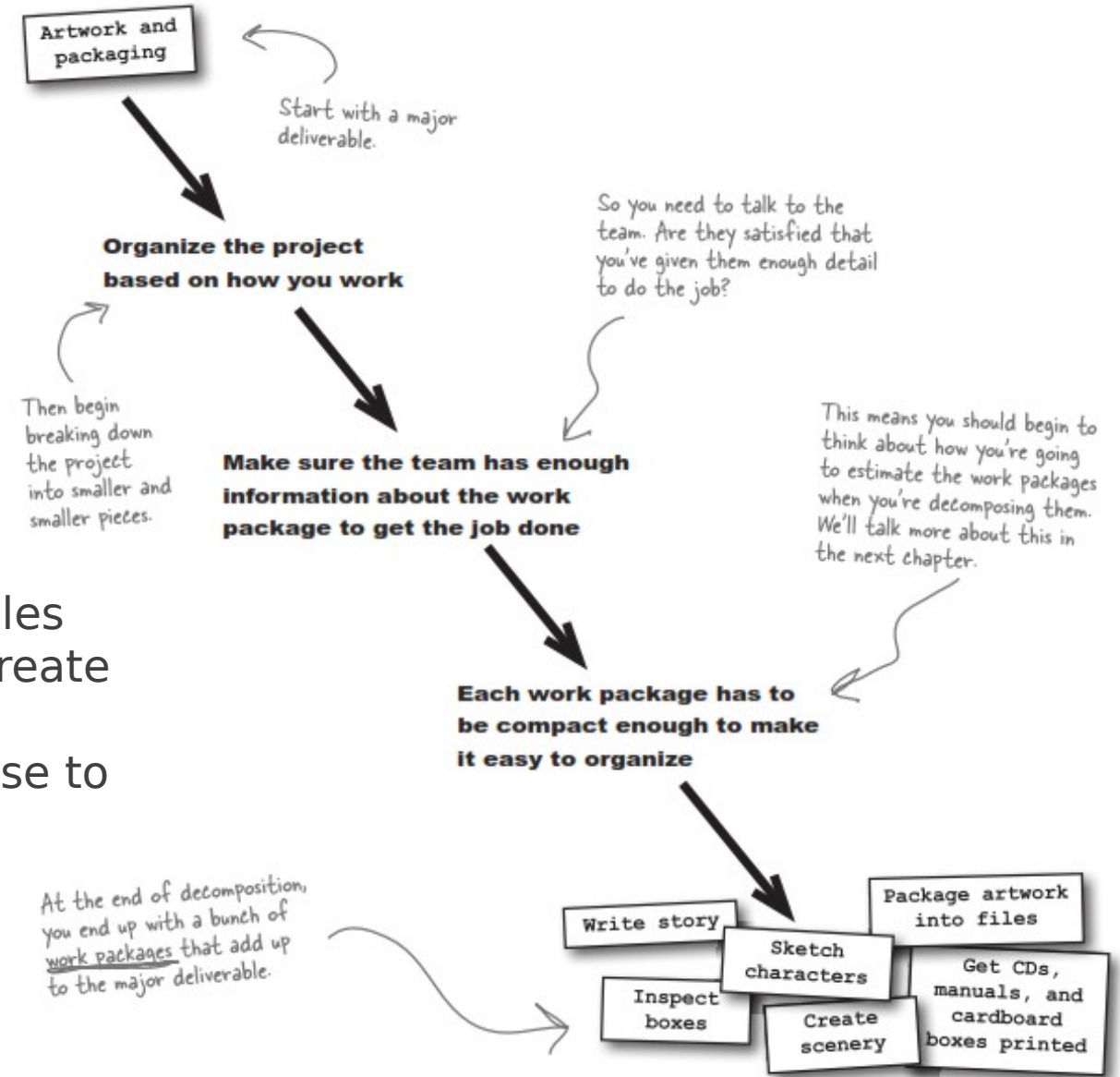
Create the work breakdown structure



The inputs for the WBS come from other processes

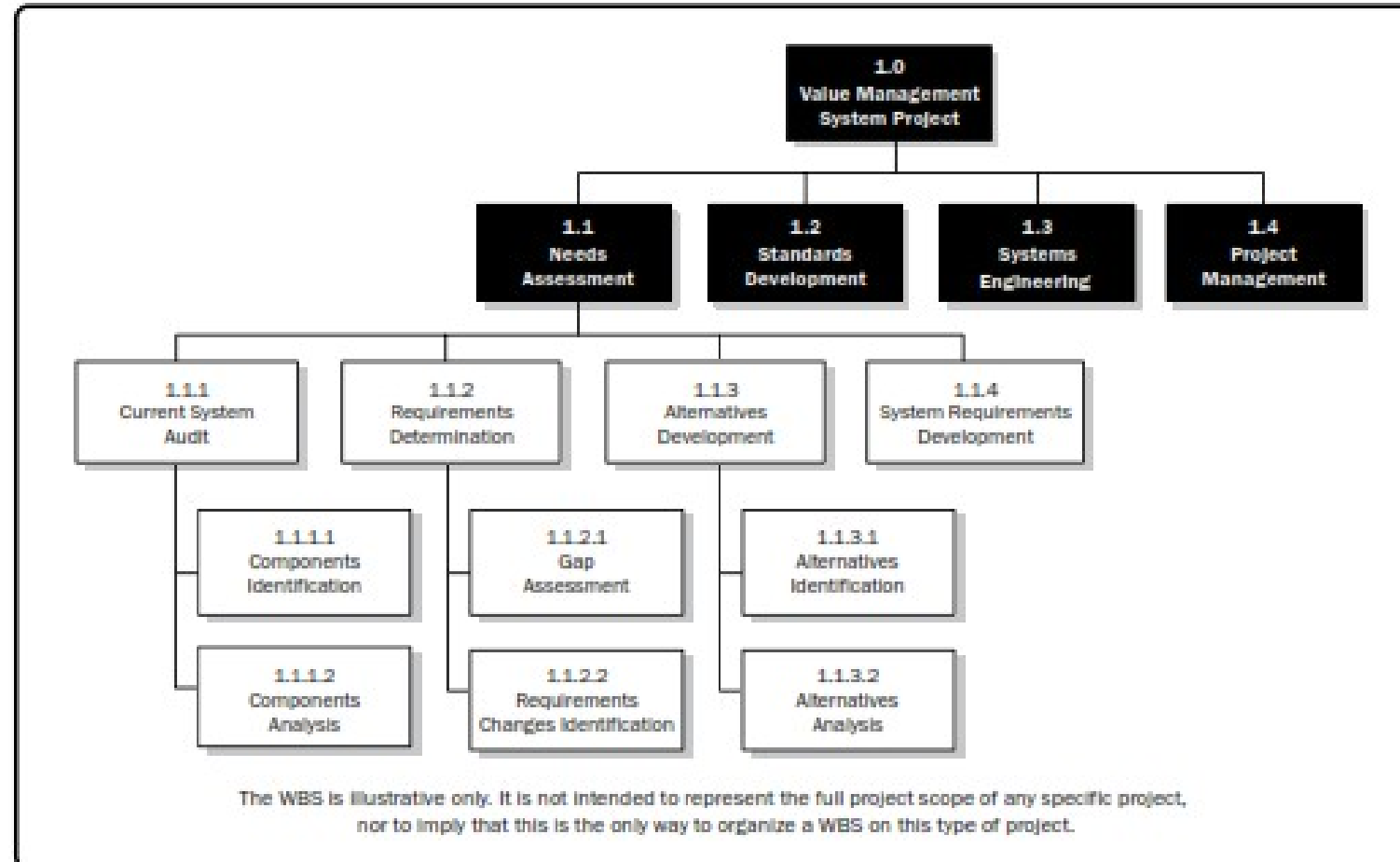


Decompose deliveral into work packages

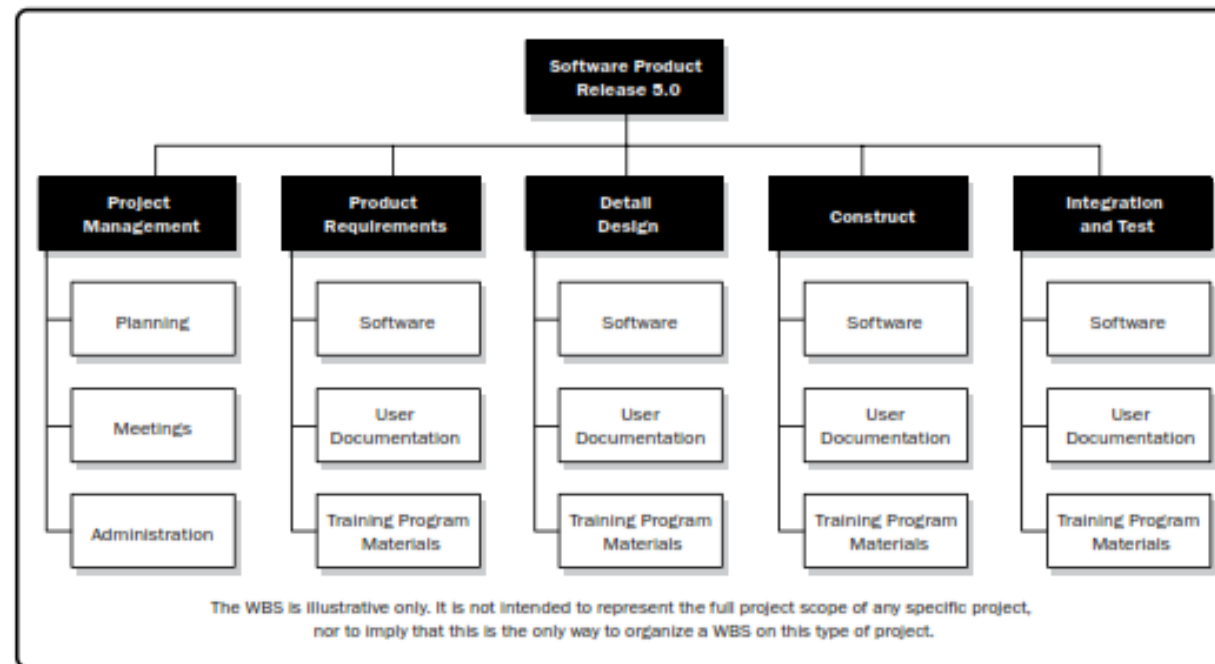


Creating the WBS is all about taking deliverables and coming up with work packages that will create them. When you do that, it's called **decomposition**, and it's the main tool you use to create a WBS.

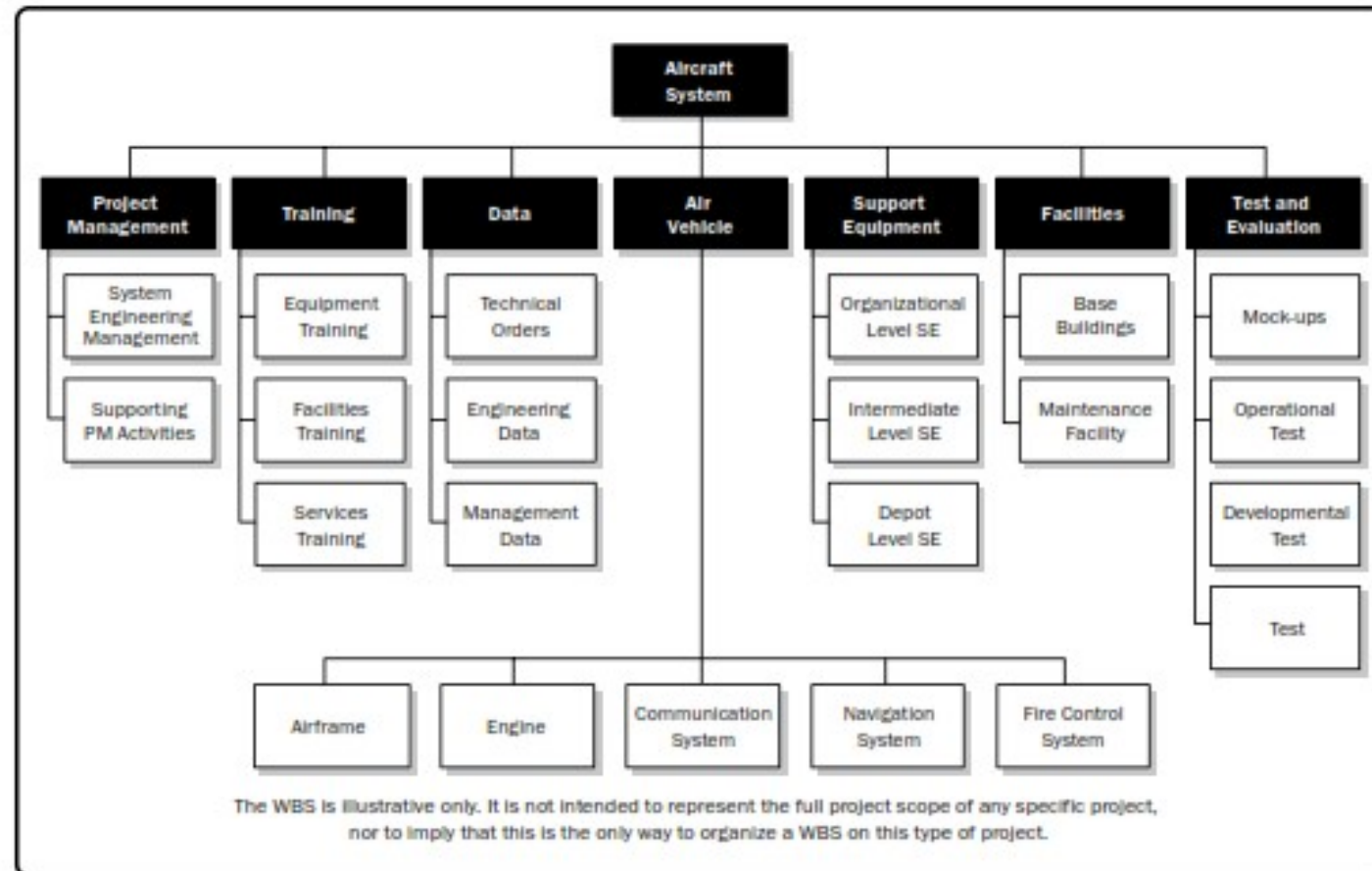
Decompose by work package



Decompose by phase



WBS with Deliverables



Inside the work package

- You've probably noticed that the work breakdown structure only shows you the name of each work package. That's not enough to do the work! You and your team need to know a lot more about the work that has to be done. That's where the **WBS dictionary comes in handy. It brings along all of the details you** need to do the project work. The WBS dictionary is an important output of the Create WBS process—the WBS wouldn't be nearly as useful without it.

The WBS dictionary contains the details of every work package. It's a separate output of the Create WBS process.

This is one of the WBS Dictionary entries for the Cows Gone Wild III project. It goes with the "Test Software" work package in the WBS.

This is just a description of the work that needs to be done.

Don't forget that the WBS doesn't show dependencies among work packages.

**Test Software
WBS Dictionary Entry**

Work Package ID and Name: 3.2.4 – Test Software

Statement of Work:
The goal of software testing is to verify that the Cows Gone Wild III software implements all of the requirements. Each requirement will be fully tested by a team of quality engineers.

Responsible Organization: Ranch Hand Games QA Team

Schedule Milestones:

- 4/26 – Programming team delivers software
- 6/18 – Functional testing and graphics testing completed
- 8/10 – QA approves software for beta testing

Quality Requirements:
The software must meet the requirements defined by the Ranch Hand Games QA team's quality standards document ("RHG QA Standards.doc")

Code of Account Identifier: RHG-236

Required Resources and Cost Estimate:

- Test planning – One QA lead and two QA analysts (\$8,500)
- Functional testing – 2 leads, 3 analysts, 11 testers (\$36,000)
- Monitor beta testing – 2 leads, 1 analyst (\$6,000)

Each work package has a name, and in many WBSes the work packages will also have ID numbers.

Here's what the WBS entry would look like with this ID number.

3.2.4 – Test Software

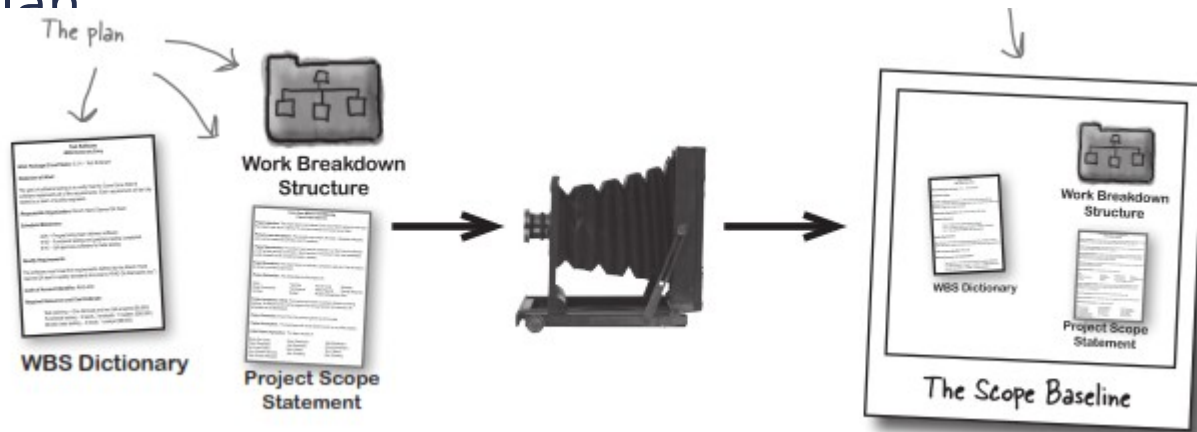
Each work package should be small enough to make cost and resource estimates.

*This account identifier is important—it's how you hook your WBS into your company's accounting system. That way, you can make sure all of the work is paid for.

The project scope baseline is a snapshot of the plan

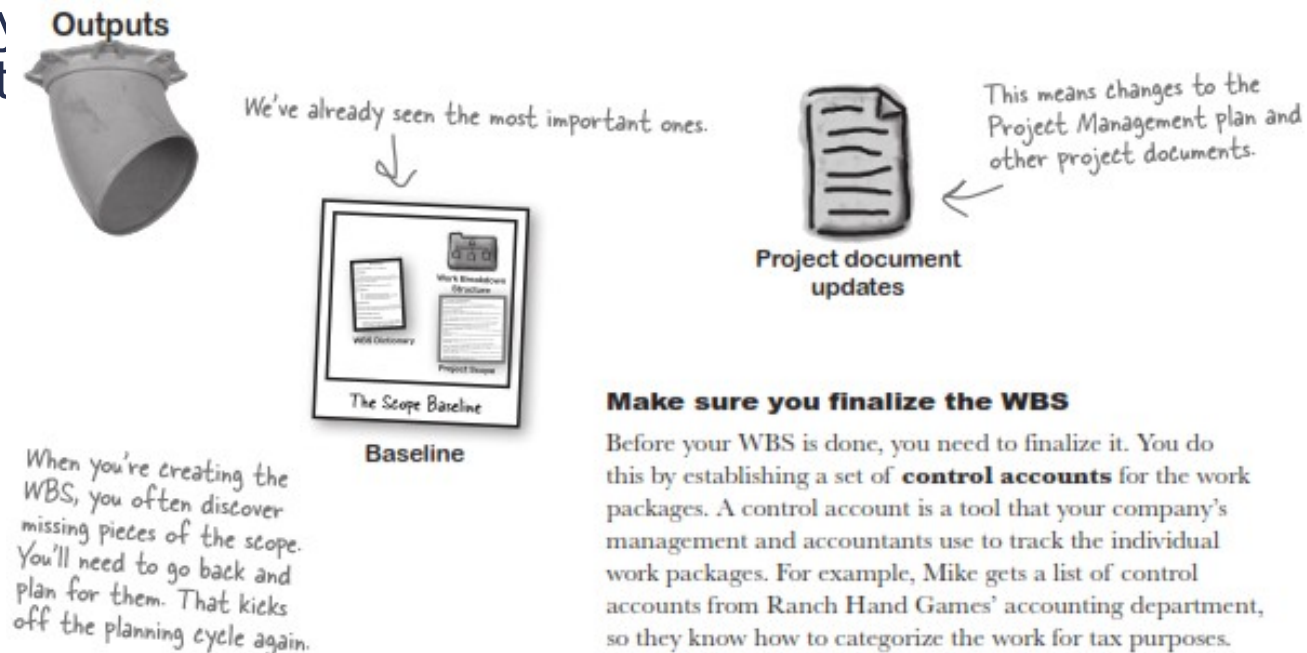
- As the project goes on, you will want to compare how you are doing to what you planned for. So, the **project scope baseline is there to compare** against. It's made up of the **project scope statement, the WBS, and the WBS dictionary**. When work gets added to the scope through change control, you need to change the baseline to include the new work packages for that work, so you can always track yourself against the plan

Any time you make a change, you need to get it approved, and then update the baseline.



The outputs of the Create WBS process

- The Create WBS process has three major outputs: the **work breakdown structure, the WBS dictionary, and the baseline**. **But there are others as well.** When you create the WBS, you usually figure out that there are pieces of the scope that you missed, and you may realize that your project document



Why scope change

The scope can change while you are working for a lot of reasons. Some changes are good for your project, while others will definitely reduce your chance of success. **Change control** is there to help you to see which is which.



Good change

A good change makes the product better with very little downside. It doesn't cost more time in the schedule or more money from the budget, and it doesn't destabilize the product or otherwise threaten its quality.

Good changes happen pretty rarely, and nearly **EVERY** change has some impact that should be fully explored before you go forward.



Bad change

A bad change is one that might seem from the outside like a good idea but ends up making an impact on the project constraints. Here are a couple of examples:

Scope creep

This happens when you think you know the impact of a change so you go ahead, but it turns out that *that* change leads to another one, and since you are already making the first change, you go with the next. Then another change comes up, and another, and another, until it's hard to tell what the scope of the project is.

The way to avoid scope creep is to plan your changes completely.

Gold plating

Sometimes people think of a really great improvement to the product and go ahead and make it without even checking the impact. In software, this can happen pretty easily. A programmer thinks of a way to make a feature better, for example, and just implements it, without talking it over with anybody. This may sound good, but it's not—because now you have to pay for these features you never asked for.

Be on the lookout for examples of scope creep and gold plating on the exam. Both are considered very bad and should never be done.

Try?

1. We need to create a screensaver to market the game. Let's kill two birds with one stone and test out a brand new graphics engine on it. Oh, and we'll need a story for the screensaver, so we should write that too. Of course we have to recruit some killer voice talent for the screensaver. Memorable names sell more games.

☐ Scope creep ☐ Gold plating

2. Testing the most recent build, I just noticed that if the player presses x-x-z-a-Shift-Shift-Space in that order, Bessie does the Charleston—it's really funny.

☐ Scope creep ☐ Gold plating

3. We should add a calculator for tracking gallons of milk collected in the game. It will be really easy. We could even release the calculator as a separate add-in, and we could probably make it full-featured enough for the folks developing the game down the hall to use it too.

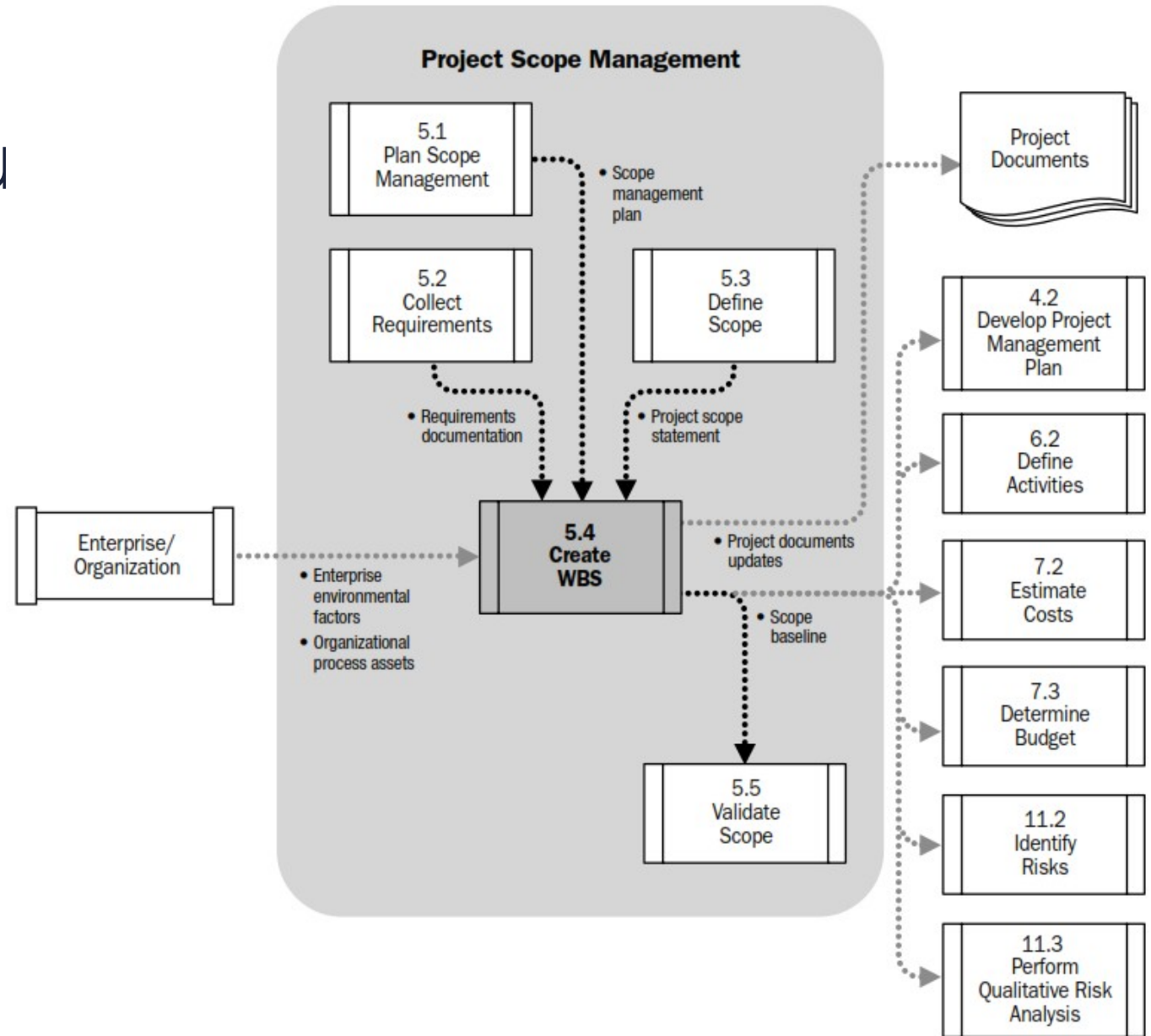
☐ Scope creep ☐ Gold plating

4. The printer just told us that she could also do silk screen T-shirts for everybody as a ship gift. Let's get our design team to do some special artwork for them. We can have everybody's names written in cows!!! Then we could use the same artwork on posters that we put around the office—oh, and coffee mugs for new people, too

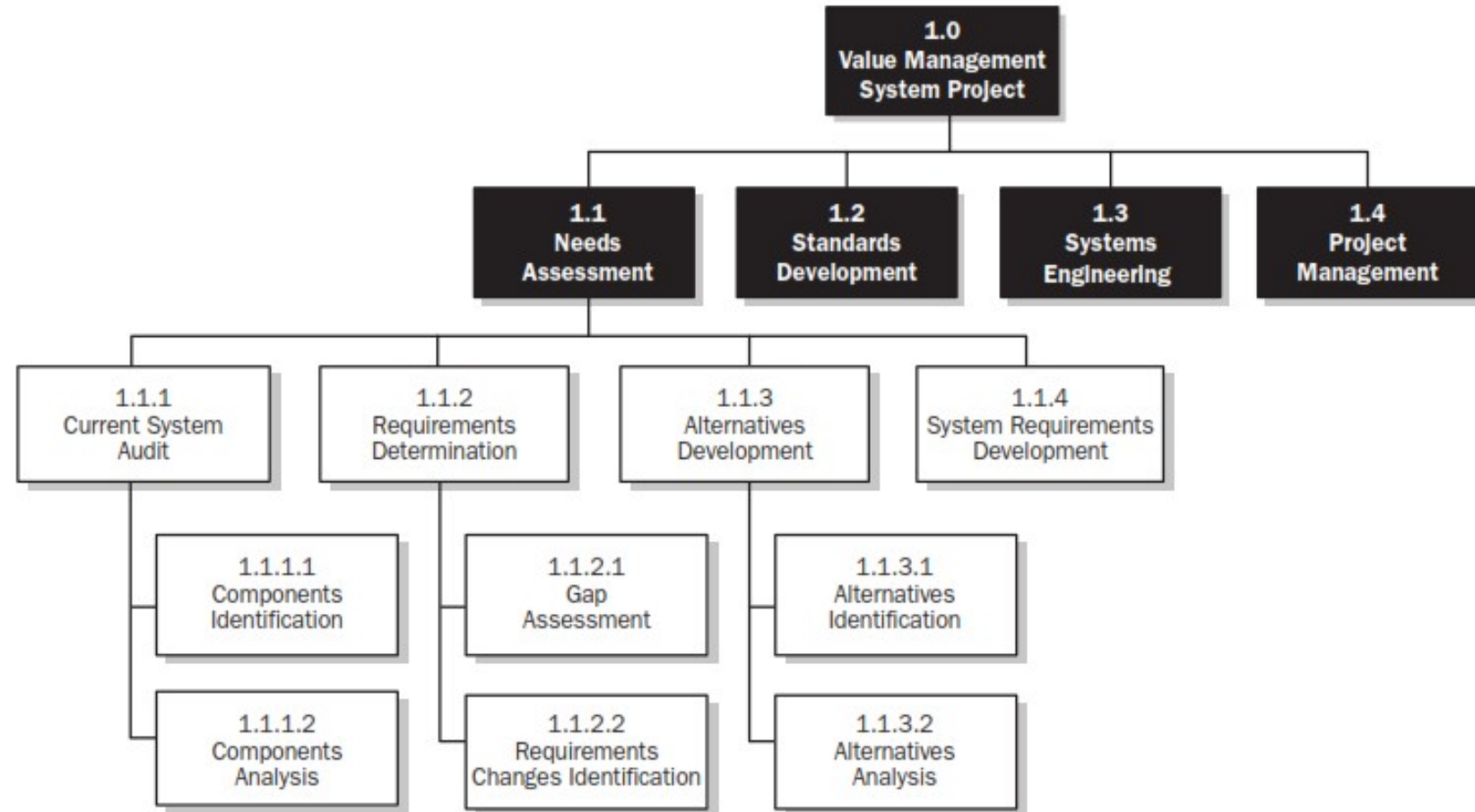
☐ Scope creep ☐ Gold plating

Answers: 1, 3, and 4 are Scope creep. 2 is Gold plating

Create WBS output



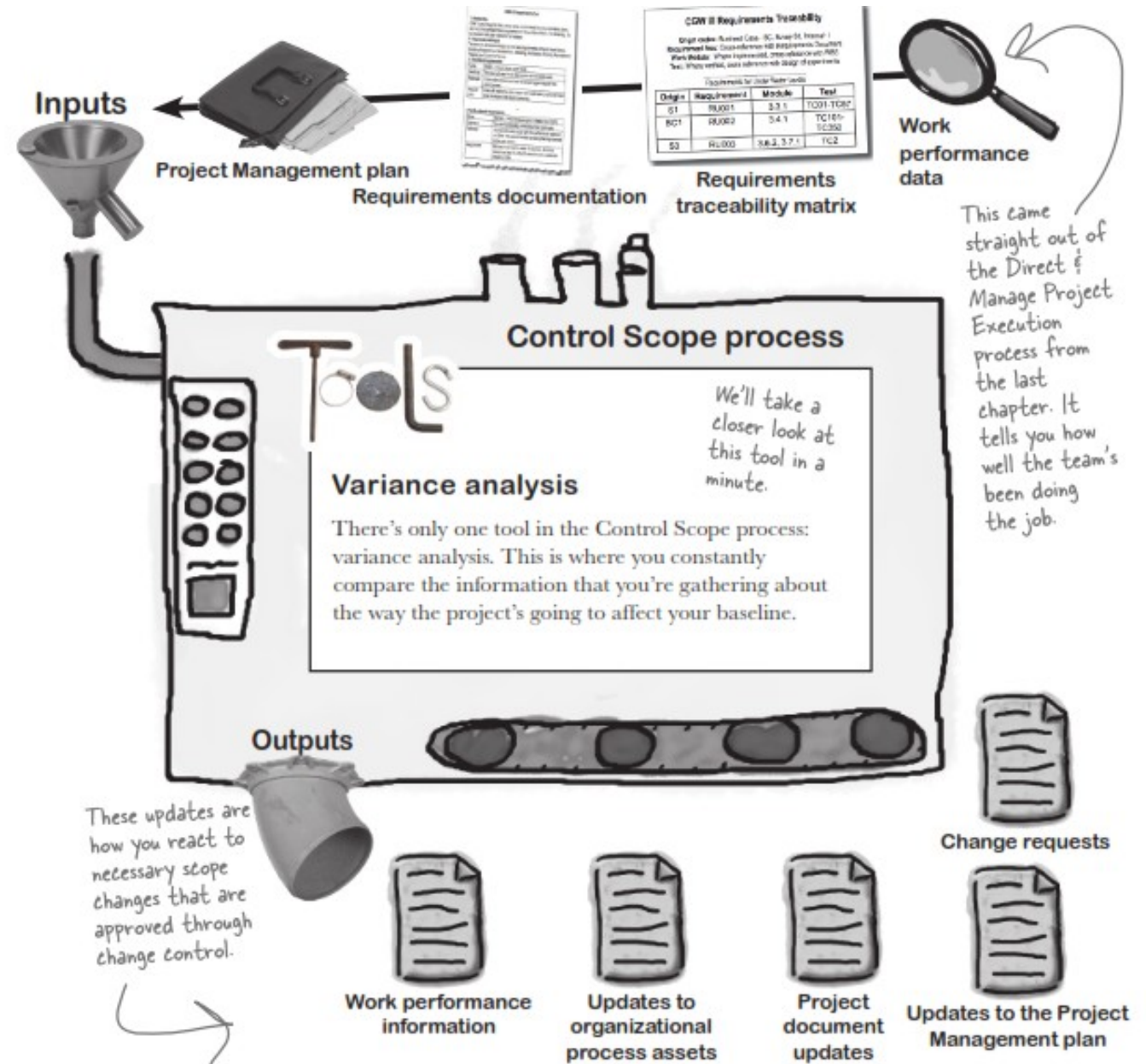
Sample WB

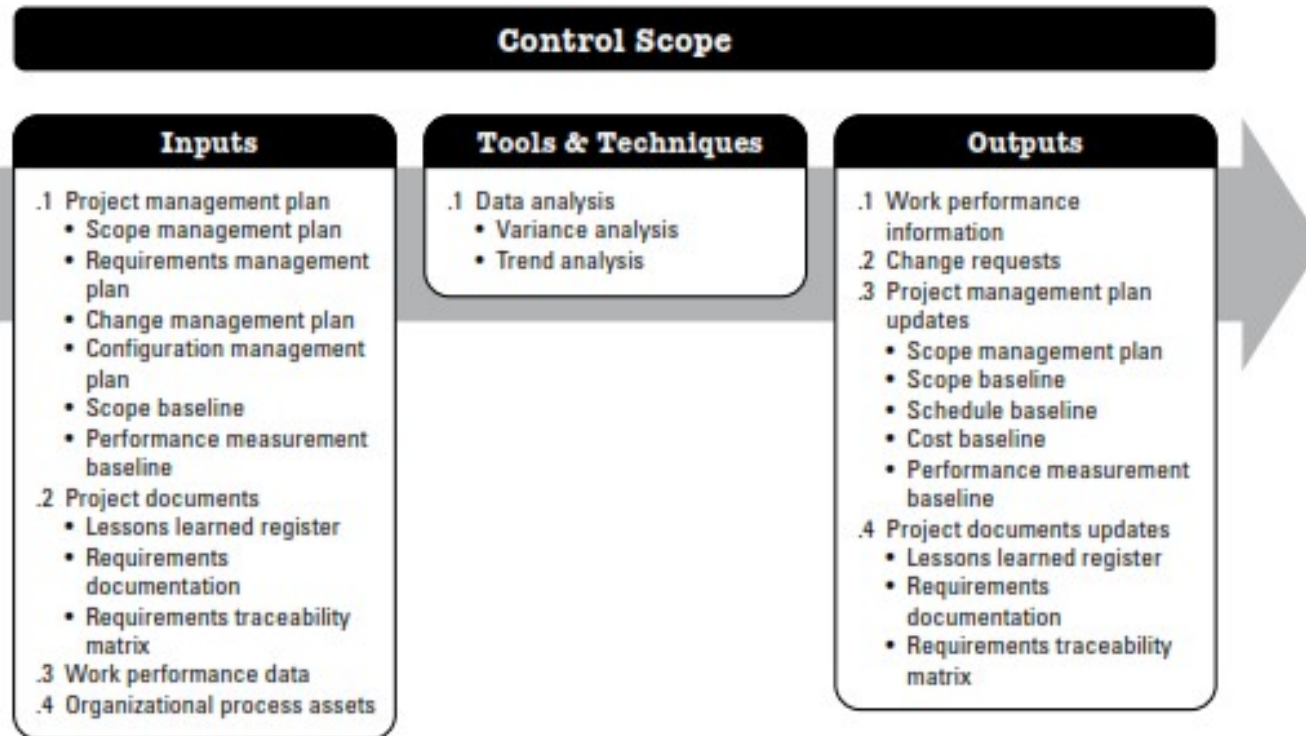


The WBS is illustrative only. It is not intended to represent the full project scope of any specific project, nor to imply that this is the only way to organize a WBS on this type of project.

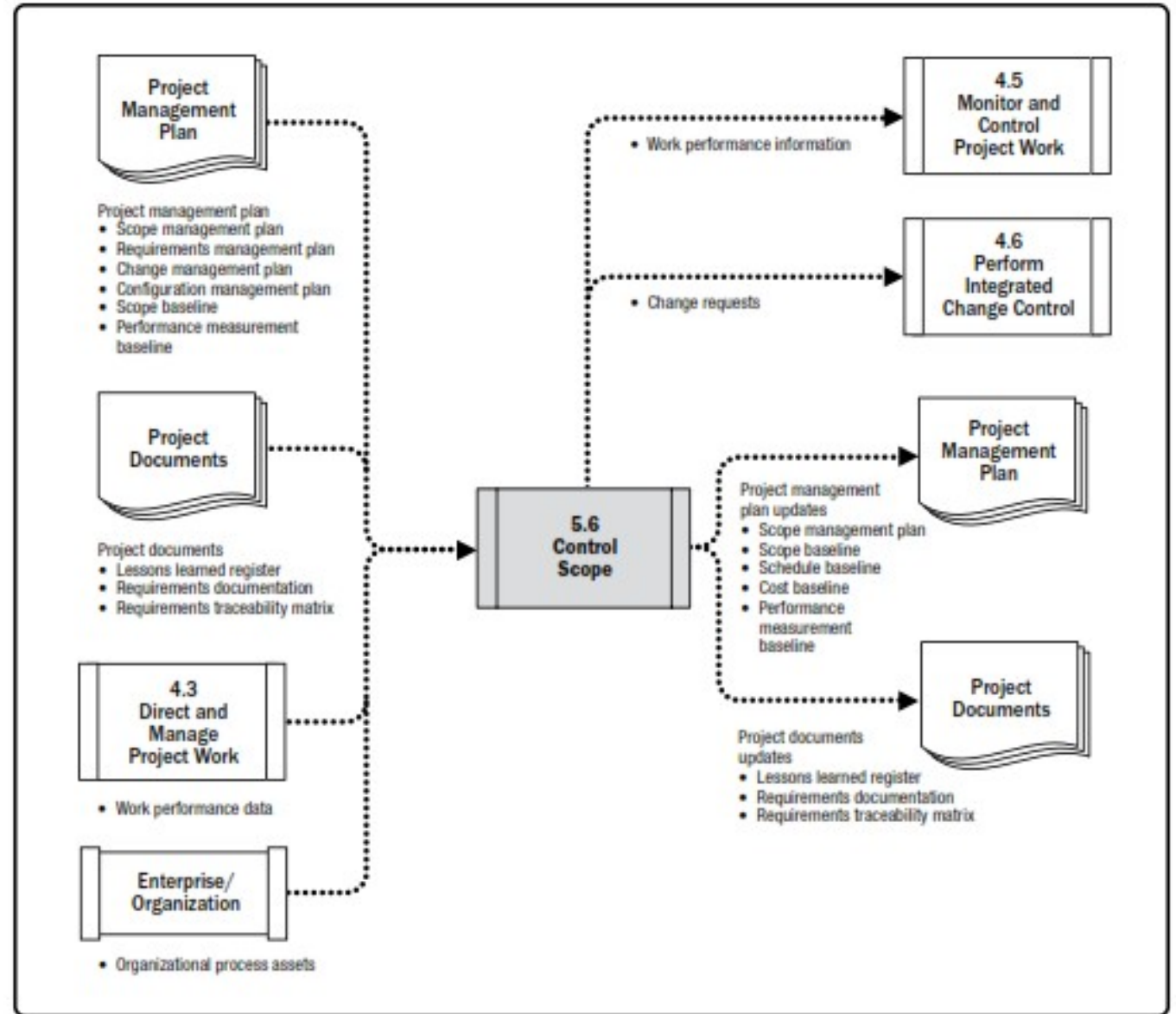
The Control Scope process

Somewhere along the way, you or someone else will realize that a **change** needs to happen, and that change will affect the **baseline**. That's why you need the **Control Scope process**. It's how you make sure that you make only those changes to the scope that you need to make, and that everyone is clear on what the consequences of those changes are.





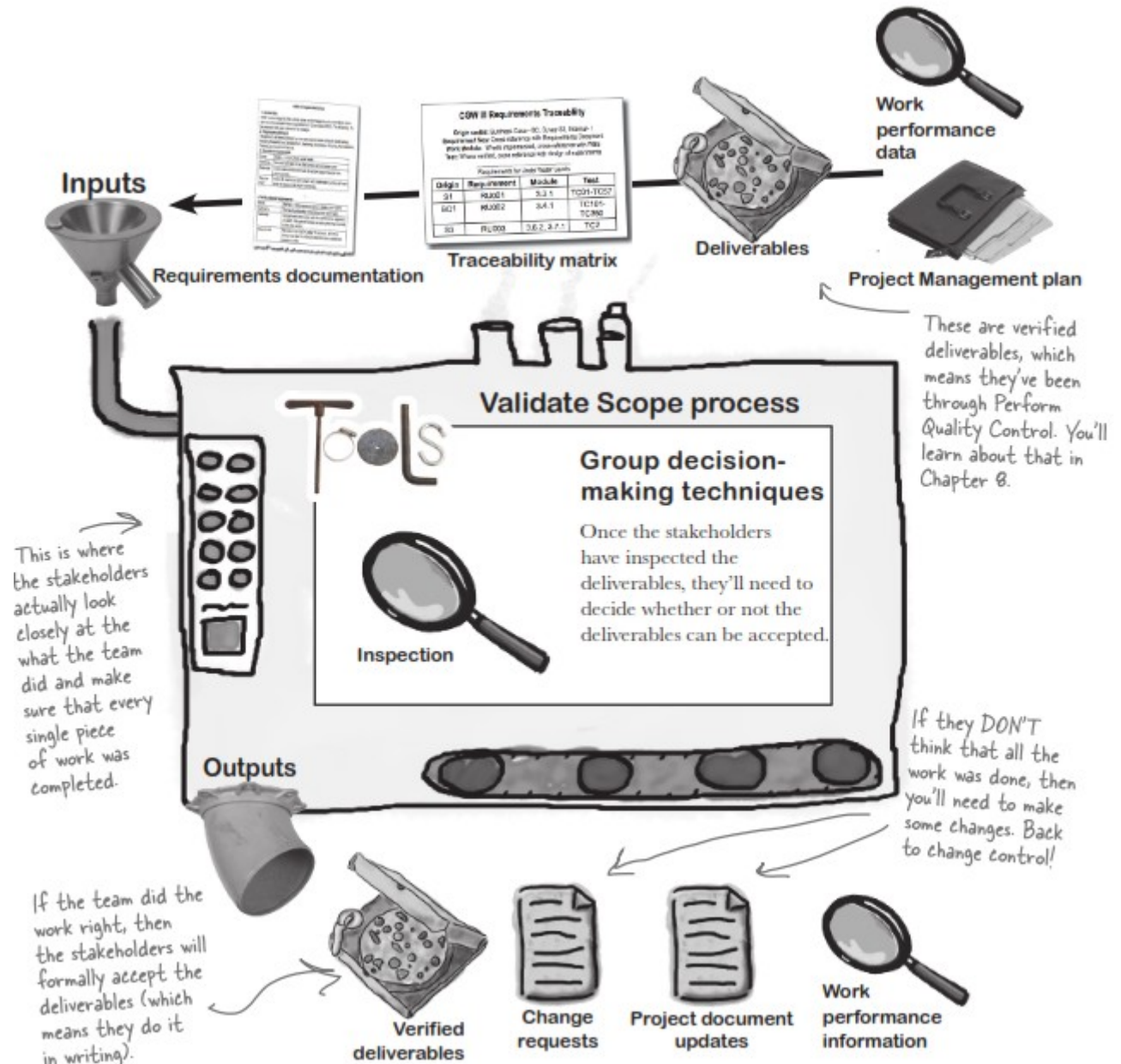
Control Scope



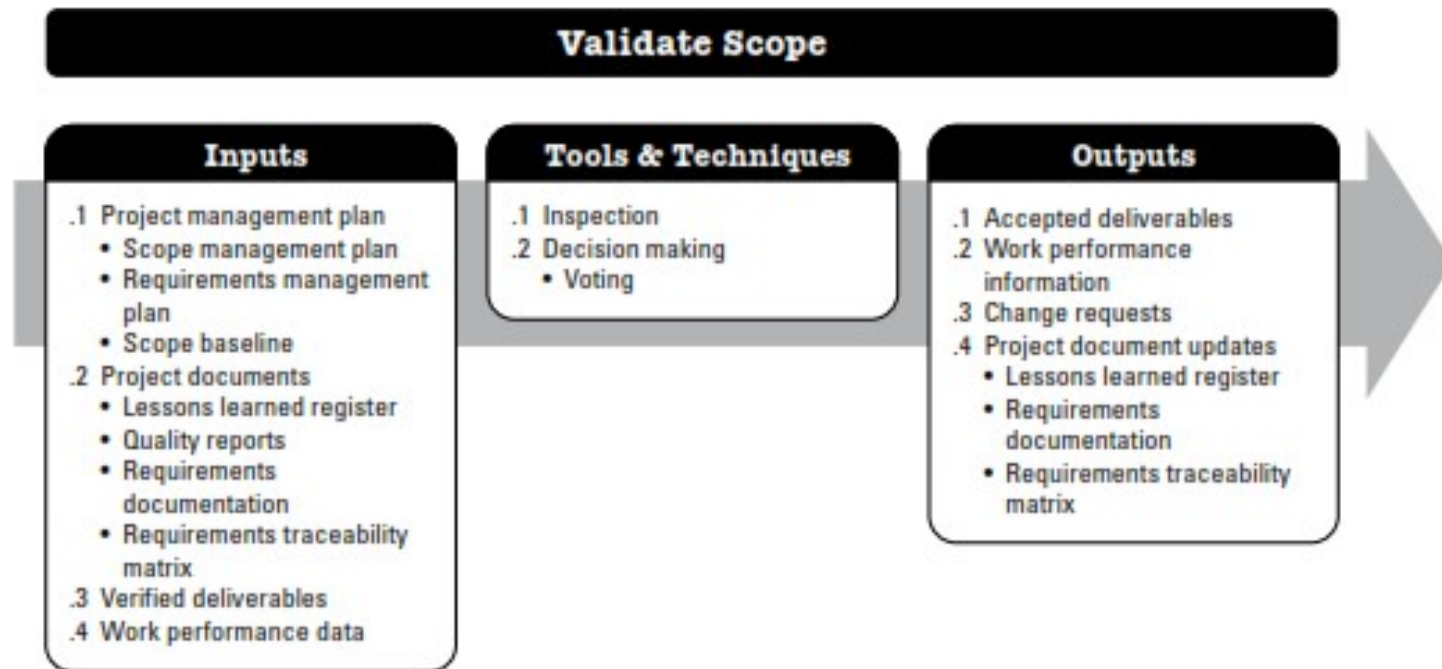
Make sure the team delivered the right product

- When the team is done, what happens? You still have one more thing you need to do before you can declare victory. You need to gather all the stakeholders together and have them **make sure that all the work really was done**. We call that the **Validate**

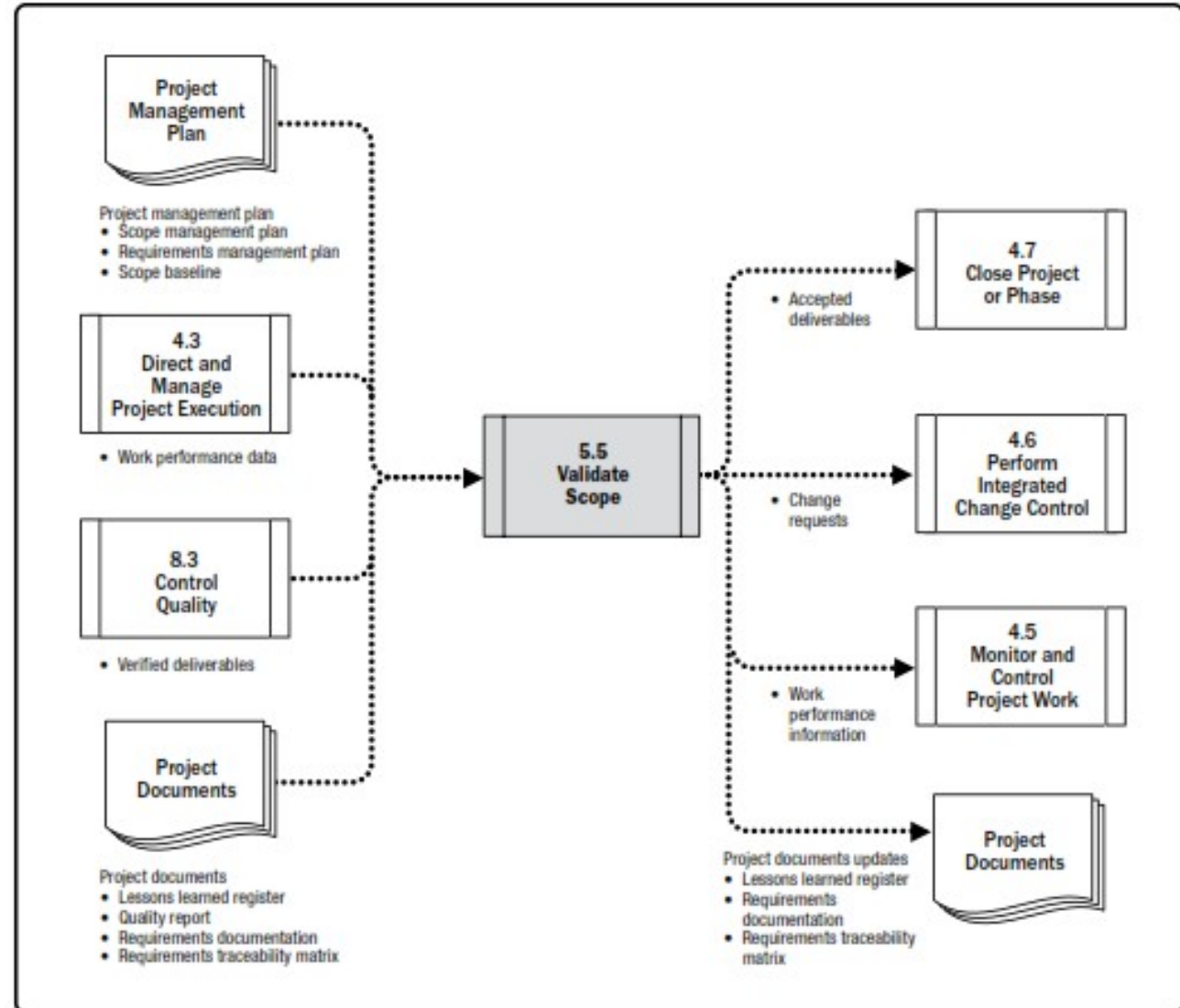
Formal acceptance means that you have written confirmation from all of the stakeholders that the deliverables match the requirements and the Project Management plan.



Validate Scope



Flow Diagram



Is the project ready to

- Once the deliverables are ready for prime time, you inspect them with the stakeholders to make sure that they meet acceptance criteria. The purpose of Validate Scope is to **obtain formal, written acceptance of the work products**. If they are found to be unsatisfactory, the specific changes requested by the stakeholders get sent to change control so that the right changes can be made.

