

Software Project Management and Planning Overview

Dr. Klyne Smith

Topics

- **Software Delivery Lifecycles**
- **Project Management**
- **Schedule Management**
- **Scope Management**
- **Risk Management**

The background features a series of concentric circles in light gray, some solid and some dashed, creating a ripple effect. A large red speech bubble is centered on the page, containing the text.

Software Delivery Life Cycle Models Project Readiness

What is a project?

A project is a planned effort with a:

- performance goal
- scheduled start
- scheduled end
- budget/resources.

It is the process of planning, applying, and controlling the use of funds, personnel, and physical resources to achieve a specific result

What are the basic project types

- **Solution Enhancement Project** – major enhancement to an existing solution
- **New Project** – customer built to meet user requirements
- **Integration Project** – integrate other software solutions into existing solution
- **Operations and Maintenance** – sustain current system, resolving user challenges, minor changes
- **Migration** – moving data and flow from one system to another

Complexity Factors

- **Technical content**
- **Cost constraints**
- **Schedule constraints**
- **Don't forget Quality...**

Steps for Categorizing a Project

1. Match the characteristics of your specific project against the project types.
2. Select the type that most closely matches your project.
3. Determine your project's technical content.
4. Analyze the project's cost and schedule parameters.
5. Consider whether the initial project type selection is appropriate, given the characteristics and complexity factors of your project.
6. If necessary, select another project type

Result:

- *Better understanding the nature of your project.*

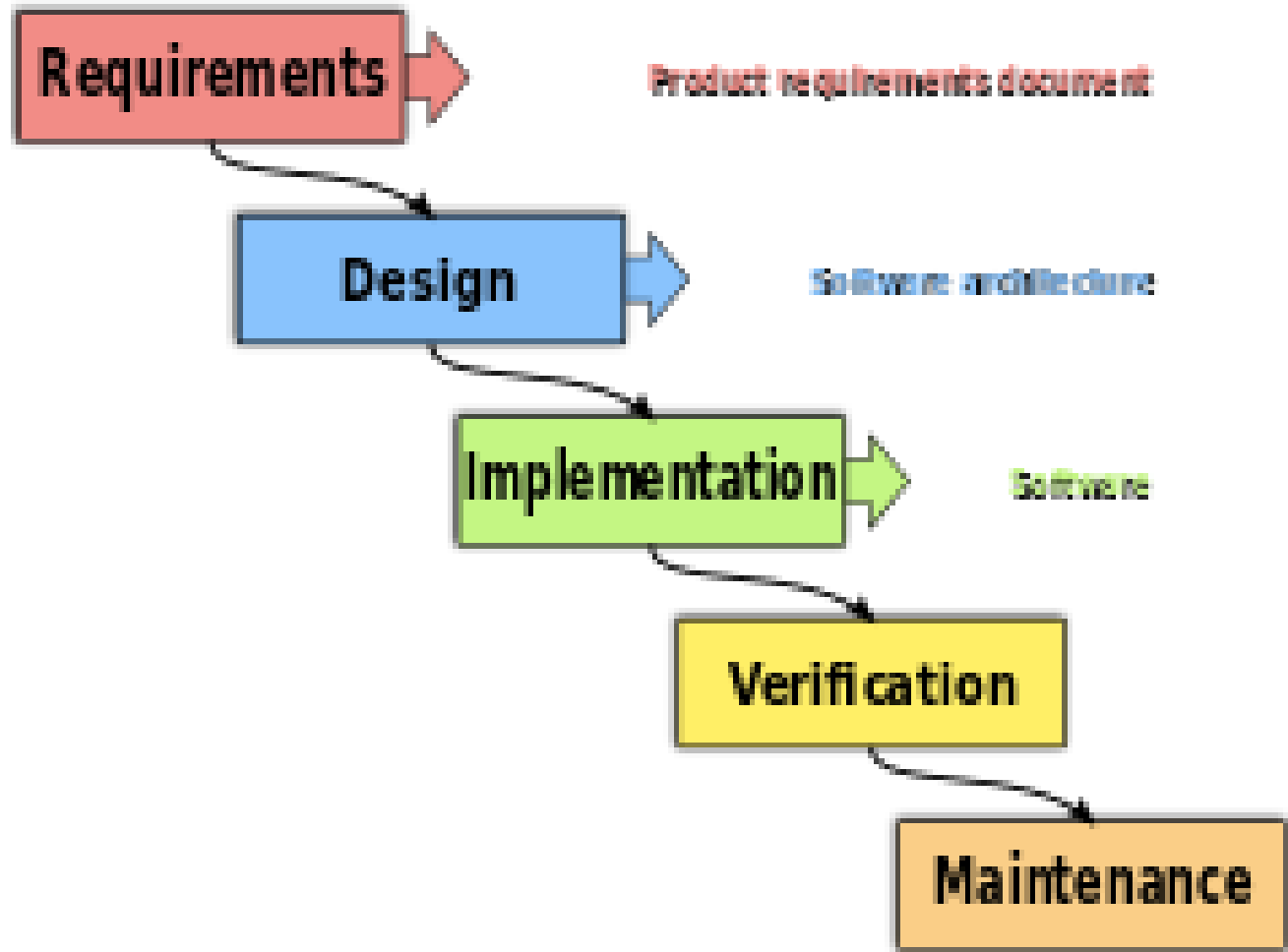
Three simple truths??

1. It is impossible to gather all the requirements at the beginning of a project
2. Whatever requirements you gather are guaranteed to change
3. There will always be more to do than time and money will allow

Waterfall

- The waterfall model runs in a sequential order.
- This steps move from higher level to lower levels traversing or falling into the next stage.
- The waterfall model is broken down into phases.
- The first phase at the topmost is inception where an individual has an idea.
- This idea leads to the initiation where measures are put in place so as to make sure that the process of development starts.
- The analysis and design then follow in order where the developer seeks to understand whether the intended model can work.
- This then trickles down to the construction where the actual development takes place.
- Next, followed by testing to find out whether the product is viable.
- The testing step is followed by implementation where the model is adopted and implemented.
- The lowest step on the ladder is to make sure that the software is maintained

Waterfall Model (Royce)



Spiral

- Due to the weaknesses of the Waterfall Model aligned with particular type of deliveries, Boehm came up with the Spiral Model.
- This model is based on the idea of a life cycle.
- This model indicates that this model took the risk-driven approach in guiding the framework for software processing.
- This seeks to make sure that the pitfalls left by the waterfall model can be minimized and addressed.
- This model is a blend of the waterfall model as well as the prototyping method.
- In relation to the waterfall model, the spiral model still has it that there are steps that need to be accomplished when seeking to develop the software.
- **However, it allows for a back flow where a mistake can be corrected.**
- The features of the model are divided into four major components. The model starts with product definition and analysis. The system and software design also have to be analyzed so as to make sure that they are in place and they cannot lead to a hitch. Lastly, there is the implementation or the code step where everything that has been agreed upon is put in place.

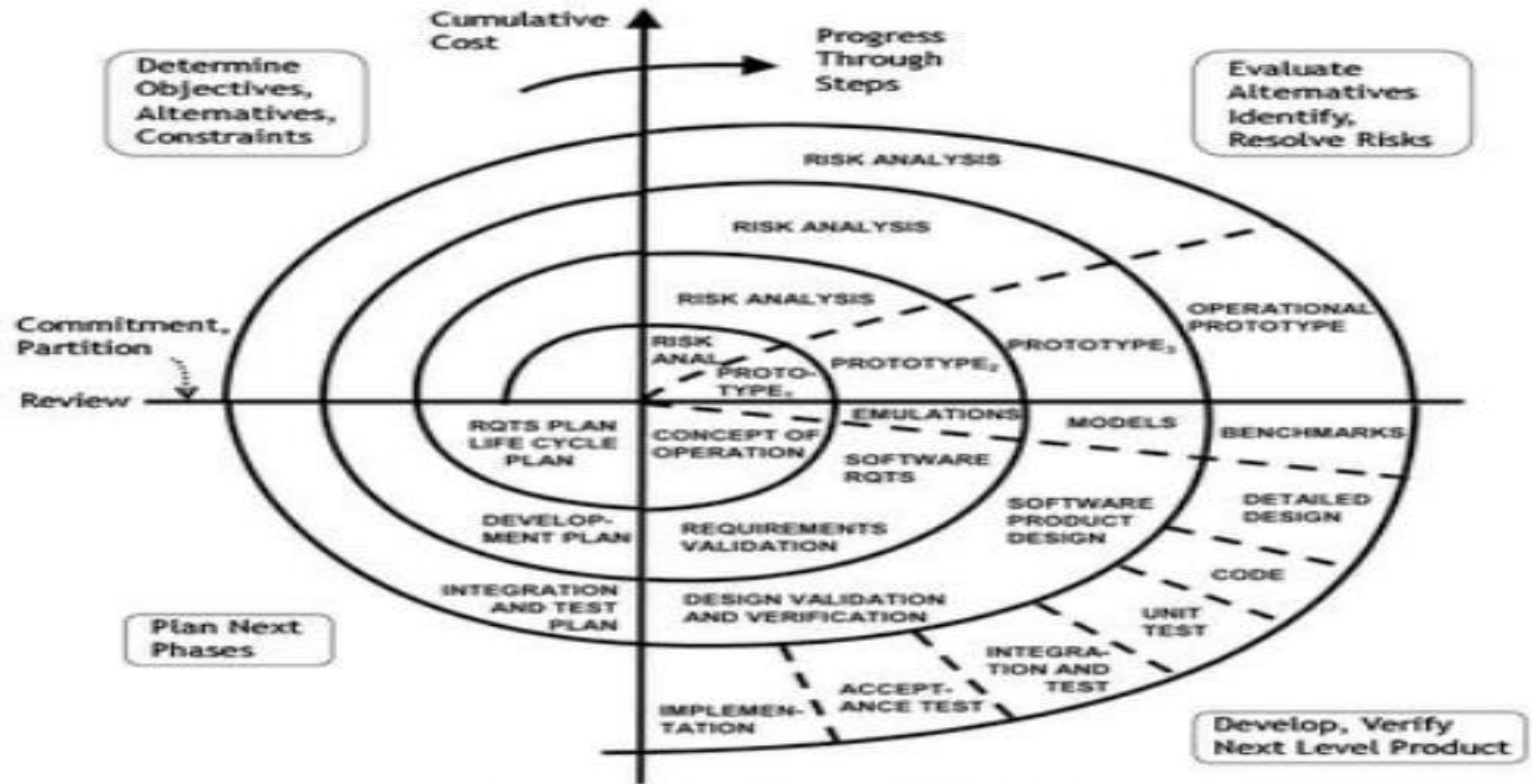
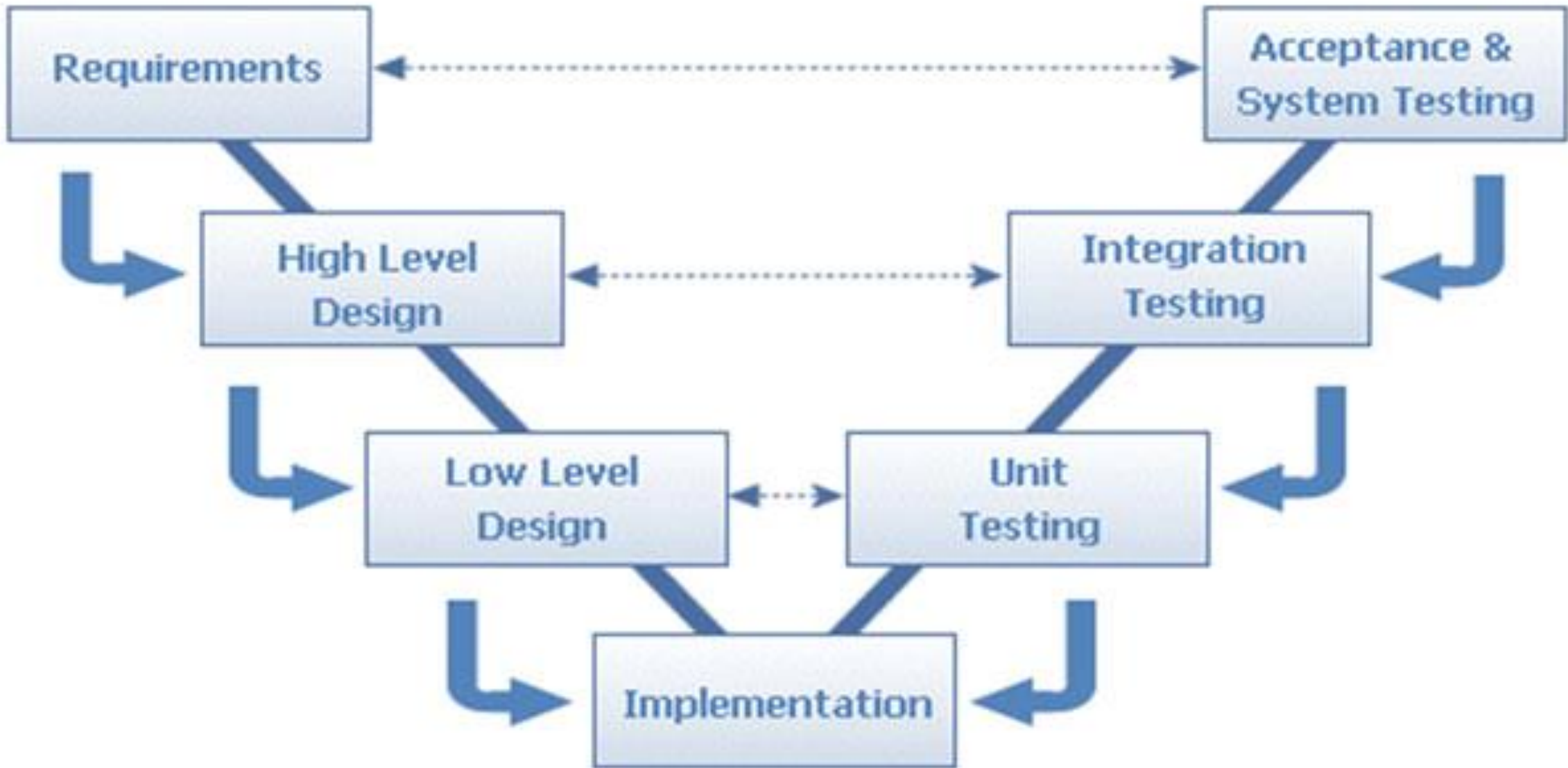


Figure 1 – Spiral Model

V-Model

- The V-Model is divided into two major sections that connect at implementation.
- There is the verification stage, which involves analysis, design, coding, module test, integration test, and system test.
- These steps are aimed at making sure that the software that comes in place is sound and ready for implementation. As such, it takes into considerations any possibility of error..
- These lie at the top edges of the V-model. The requirements section indicates that the developers have to have in mind a specific objective that has to be fulfilled by the program.
- Focusing is the main objective of this model. The acceptance test implies that the final product has to be assessed so as to make sure that it satisfies the need for which it was created.
- This step helps to make the developers keen in their development activities. This is mainly because failure to follow the steps will lead to rejection of the product.
- Based on this model, it can be argued that software processing has to be aligned to the needs of the developer, the type of project, and current working environment.



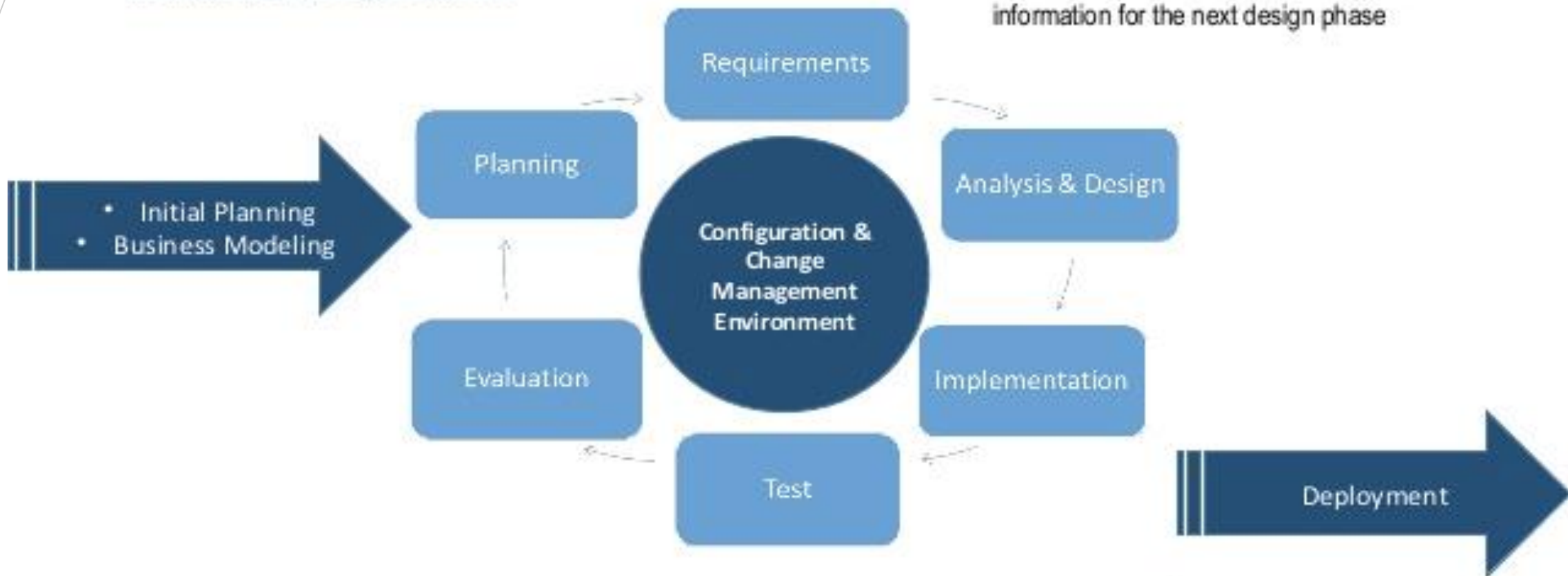
Iterative Model

- The Iterative model has it that the software developers have to have an idea of the final product they want to achieve.
- This is also known as prototyping. However, they need to produce an example of the software so as to test whether it is functional or not.
- This is the prototyping step. If the prototype is accepted and proven to be viable, then the developers engage in an activity to produce more copies of the same and implement them.
- The main idea behind this model is that before formalizing a process, all the important components of the project have to be defined and illustrated in a model. The model helps to determine whether the software is valid or not, and whether it should be accepted.

Iterative model

incremental life cycle model

Each iteration is a mini-waterfall process, with the feedback from one phase providing vital information for the next design phase



Before
understanding
the agile model....

What is agile?

Agile is a time boxed, iterative approach to software delivery that builds software incrementally from the start of the project, instead of trying to deliver it all at once near the end.

Agile software development refers to a group of software development methodologies based on iterative development, where requirements and solutions evolve through collaboration between self-organizing cross-functional teams.

Agile processes generally promote a disciplined project management process that encourages frequent inspection and adaptation, a leadership philosophy that encourages teamwork, self-organization.

User stories are features our customers might one day like to see in their software (these are not requirements)

How does it work?

- 1) Make a list
- 2) Estimate
- 3) Prioritize
- 4) execute

Master story list

Add user
Print itinerary
Cancel trip
...
Search
Create device
Add swap trade

Master story list

1 day Add user
2 days Print itinerary
1 day Cancel trip
... ..
5 days Search
5 days Create device
3 days Add swap trade
100 days

Master story list

1 day Add user
1 day Cancel trip
2 days Print itinerary
... ..
5 days Search
3 days Add swap trade
5 days Create device

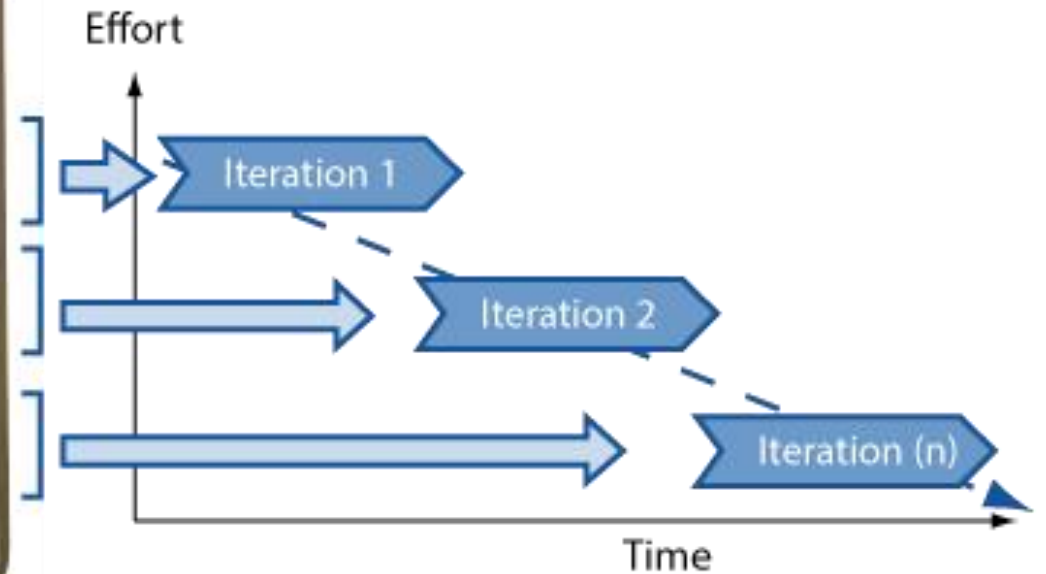
Master story list

1 day Add user
1 day Cancel trip
2 days Print itinerary
... ..
5 days Search
3 days Add swap trade
5 days Create device
100 days

Agile delivery mindset



Master story list	
1 day	Add user
2 days	Print itinerary
5 days	Cancel trip
3 days	Book permit
1 day	Update permit
...	...
5 day	Create device
3 day	Add swap trade
<hr/>	
100 days	



What is Scrum?

- Scrum is a subset of Agile.
- It is a lightweight process framework for agile development, and the most widely-used.
- A “process framework” is a particular set of practices that must be followed in order for a process to be consistent with the framework. (For example, the Scrum process framework requires the use of development cycles called Sprints, the XP framework requires pair programming, and so forth.)
- “Lightweight” means that the overhead of the process is kept as small as possible, to maximize the amount of productive time available for getting useful work done.

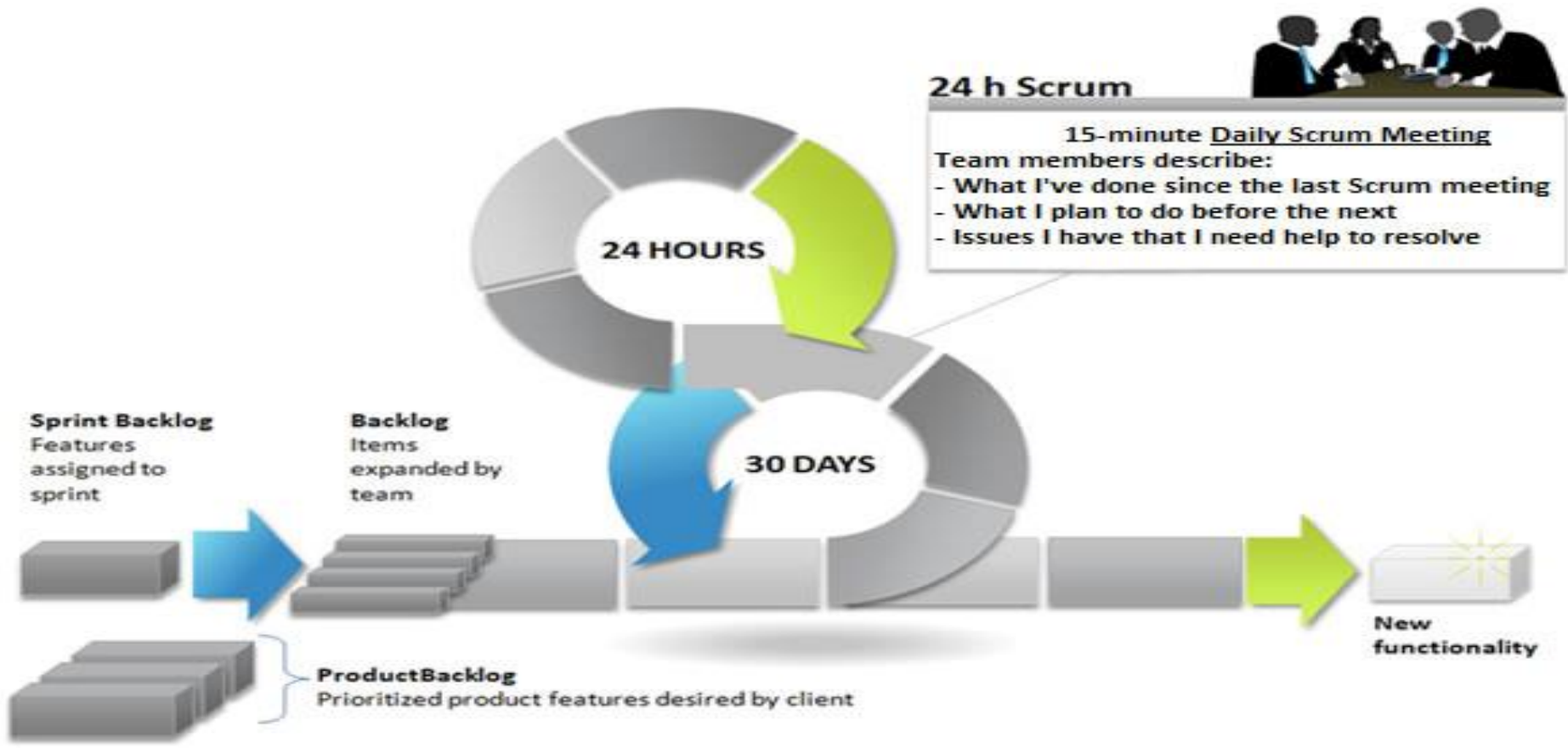
What is a scrum process?

- A Scrum Process is distinguished from other agile processes by specific concepts and practices, divided into the three categories of Roles, Artifacts, and Time Boxes.
- Scrum is most often used to manage complex software and product development, using iterative and incremental practices.
- Scrum significantly increases productivity and reduces time to benefits relative to classic “waterfall” processes.
- Scrum processes enable organizations to adjust smoothly to rapidly-changing requirements, and produce a product that meets evolving business goals.
- An agile Scrum process benefits the organization by helping it to
 - Increase the quality of the deliverables
 - Cope better with change (and expect the changes)
 - Provide better estimates while spending less time creating them
 - Be more in control of the project schedule and state

Agile – scrum key roles

- **Project Manager** – overall lead for the project accountable for success and meeting project objectives
- **Product Owner** – a product owner represents the product (business). They analyze market and business trends. They set the priority of activities for the Scrum team
- **Scrum Master** – leads team following Scrum principles. Educating product owners, project manager, and development team. Development overall lead. Communicates key information to Project Manager.
- **Development Team** - usually five to seven members focusing on effective development methodologies. Each member has a unique set of skills, i.e. developer, architect, testing, etc.

SCRUM PROCESS



The image features a large red speech bubble with a white outline, pointing downwards. Inside the bubble, the words "Project Management" are written in a white, sans-serif font. The background is white with faint, light gray concentric circles and dashed lines, creating a subtle pattern.

Project Management

Control Gates

A means of ensuring that
the required work
has been satisfactorily accomplished
before proceeding to the next phase of a project.

Control Gate Objectives

- Provide progressive Project Control throughout the Project Cycle.
- Validate progress by examining evidence of technical, schedule and cost performance.
- Avoid progressing to a subsequent activity for which the project is not prepared.
- Promote a synergistic problem solving environment among team members.

Project Management Techniques

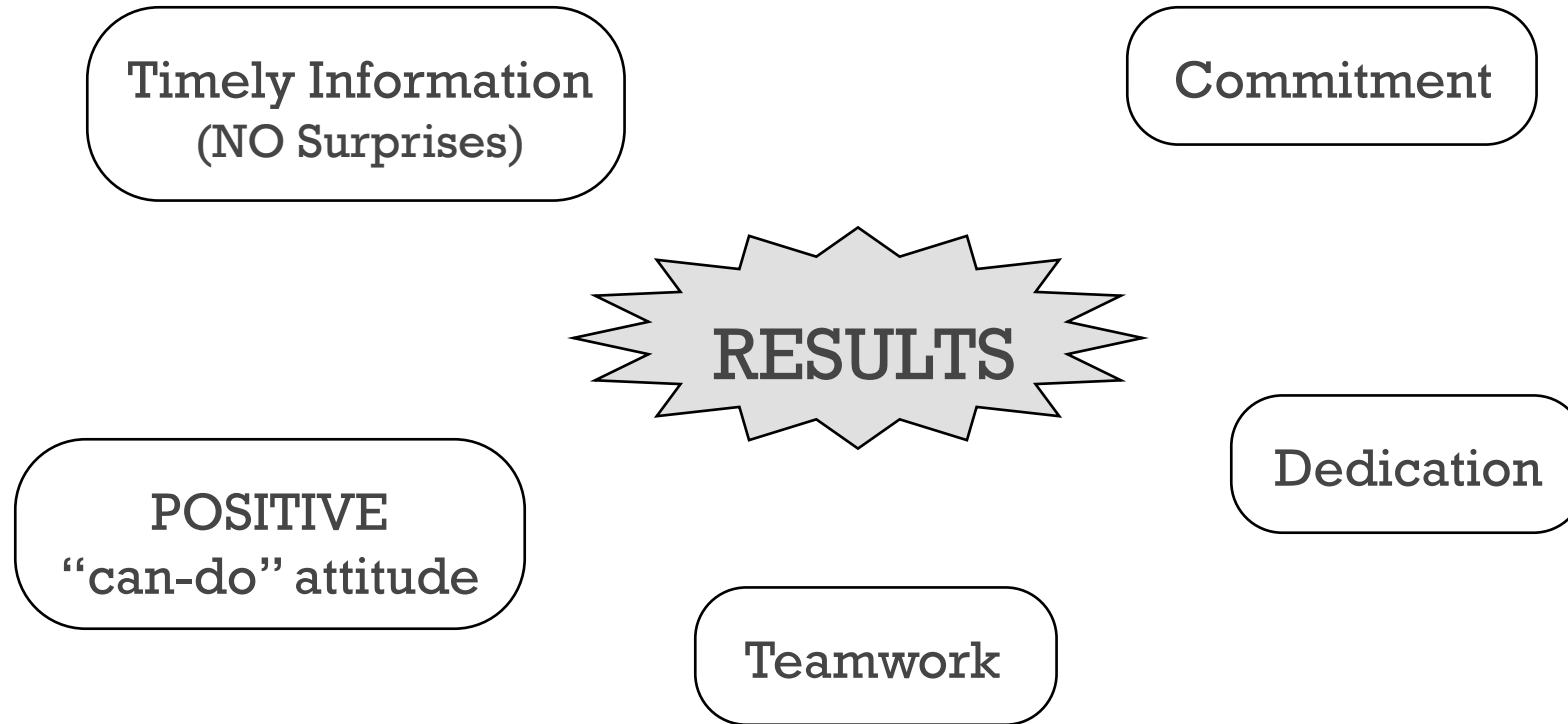
- **Ten areas provide techniques and tools that can be applied by the team throughout the life of the project:**
 - **Leadership – leading the team**
 - **Project Management – selecting & training the PM**
 - **Project Visibility – knowing what is going on**
 - **Project Requirements Traceability – managing requirements**
 - **Project Planning – planning the work**
 - **Risk Management – managing the impact of risk**
 - **Project Organization – organizing the team**
 - **Project Control – controlling deviations from plan**
 - **Project Status – measuring progress**
 - **Corrective Action – implementing effective actions**

What The Customer/User Expects From The Project Manager



Interactive management, punctuated by responsiveness.

What the Project Manager Expects of the Team



The Project Manager Is Responsible for Creating an Effective Team

- Foster the four essentials of teamwork.
- Inspire the team to achieve a vision.
- Set an example:
 - Demonstrate customer orientation.
 - Be honest and objective.
 - Create an environment in which problems are surfaced and risks are acknowledged.
 - Practice urgency management.
 - Be willing to do what the team is asked to do.
 - Manage the business aspects, as well as the technical aspects of the project.

Project Manager Attributes

- **Enthusiastic self-starter with a positive, "can-do" attitude**
- **Objective and balanced outlook**
- **Determination and persistence to satisfy the project objectives**
- **A management attitude that encourages others to develop recommendations and decisions that are logical and supportable**
- **A personal attitude of integrity, fairness, and trust**
- **A sense of proper closure**

Project Manager's Skills Inventory

- **Prior successful task management experience**
- **Prior successful administrative experience**
- **Understanding of the technology required**
- **Understanding of the User/Customer**
- **Credibility with the User/Customer**
- **Planning skills**
- **Leadership skills**
- **Interpersonal skills**
- **Team building capability**

Project Manager's Skills Inventory

(Continued)

- Project management training
- Systems engineering training
- Systems engineering experience
- Project business management training
- Project business management experience
- Proposal evaluation experience
- Source Selection planning experience
- Decision analysis experience
- Problem solving skills

The background features a series of concentric circles in light gray, some solid and some dashed, creating a ripple effect. A prominent red callout box is centered on the page, containing the text 'Schedule Management'.

Schedule Management

The background features a series of concentric circles in light gray, some solid and some dashed, creating a ripple effect. A prominent red callout box is centered on the page, containing the text 'Scope Management Plan'.

Scope Management Plan

What is Scope management?

- **Project Scope Management includes the processes required to ensure that the project includes all the work required, and only the work required, to complete the project successfully.**
- **Managing the project scope is primarily concerned with defining and controlling what is and is not included in the project**

What is a Scope management plan?

- **Plan Scope Management is the process of creating a scope management plan that documents how the project scope will be defined, validated, and controlled.**
- **The key benefit of this process is that it provides guidance and direction on how scope will be managed throughout the project.**
- **The key inputs are the Project Management Plan and Project Charter**

What is in a Scope management plan?

The scope management plan is a component of the project or program management plan that describes how the scope will be defined, developed, monitored, controlled, and verified. The scope management plan is a major input into the Develop Project Management Plan process, and the other scope management processes. The components of a scope management plan include:

- Process for preparing a detailed project scope statement;
- Process that enables the creation of the WBS from the detailed project scope statement;
- Process that establishes how the WBS will be maintained and approved;
- Process that specifies how formal acceptance of the completed project deliverables will be obtained; and
- Process to control how requests for changes to the detailed project scope statement will be processed.
- This process is directly linked to the Perform Integrated Change Control process

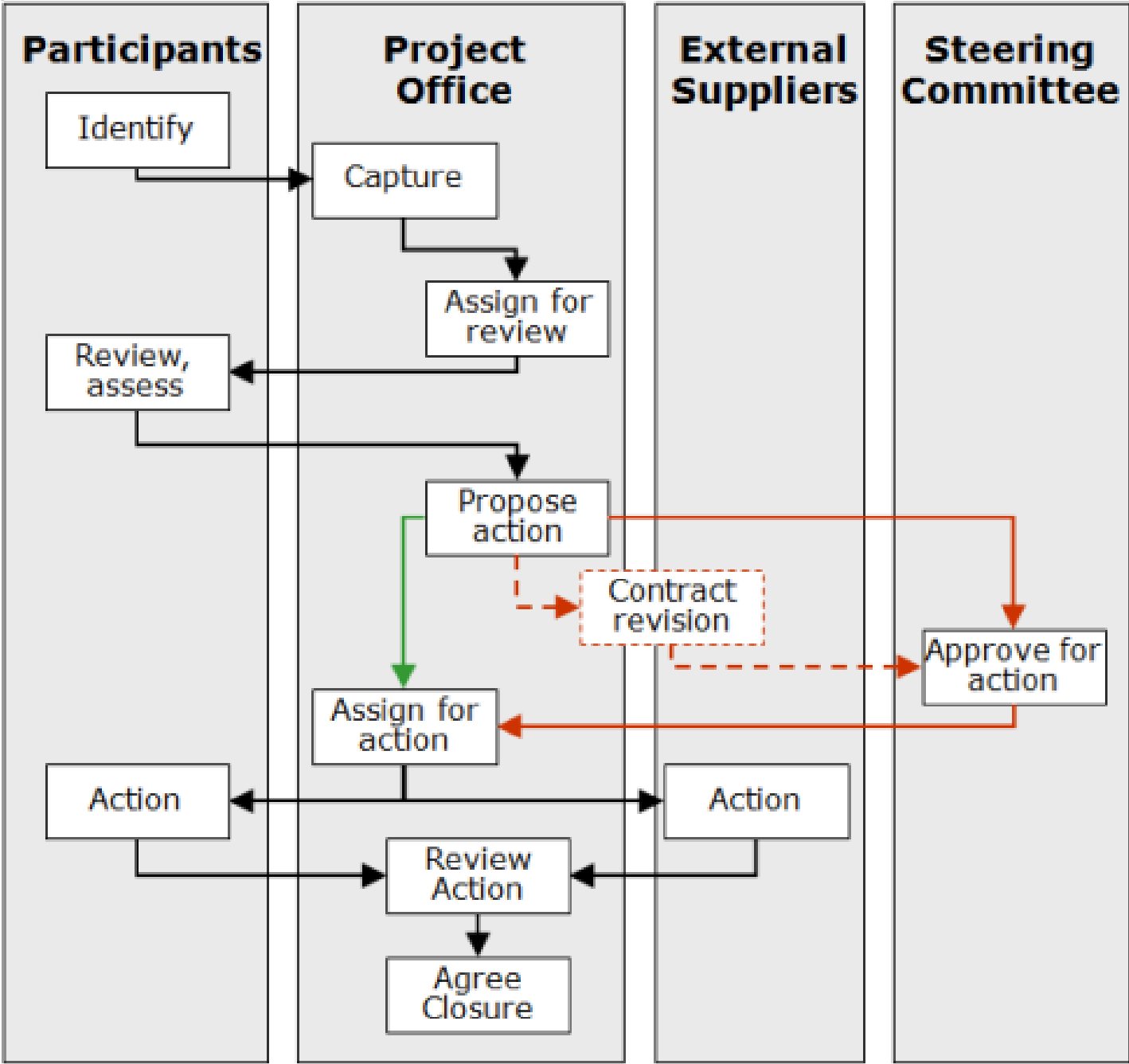
Change control process

- All of the Monitoring and Controlling processes and many of the Executing processes produce change requests as an output.
- Change requests may include corrective action, preventive action, and defect repairs. However, corrective and preventive actions do not normally affect the project baselines—only the performance against the baselines
- Change control procedures, including the steps by which official organization standards, policies, plans, and other project documents will be modified, and how any changes will be approved, validated, and implemented;
- Procedures for approving and issuing change authorizations; Process measurement database used to collect and make available measurement data on processes and products;
- Project documents (e.g., scope, cost, and schedule baselines, project calendars, project schedule network diagrams, risk registers, planned response actions, and defined risk impact); and
- Configuration management knowledge base containing the versions and baselines of all official organization standards, policies, procedures, and any project documents.

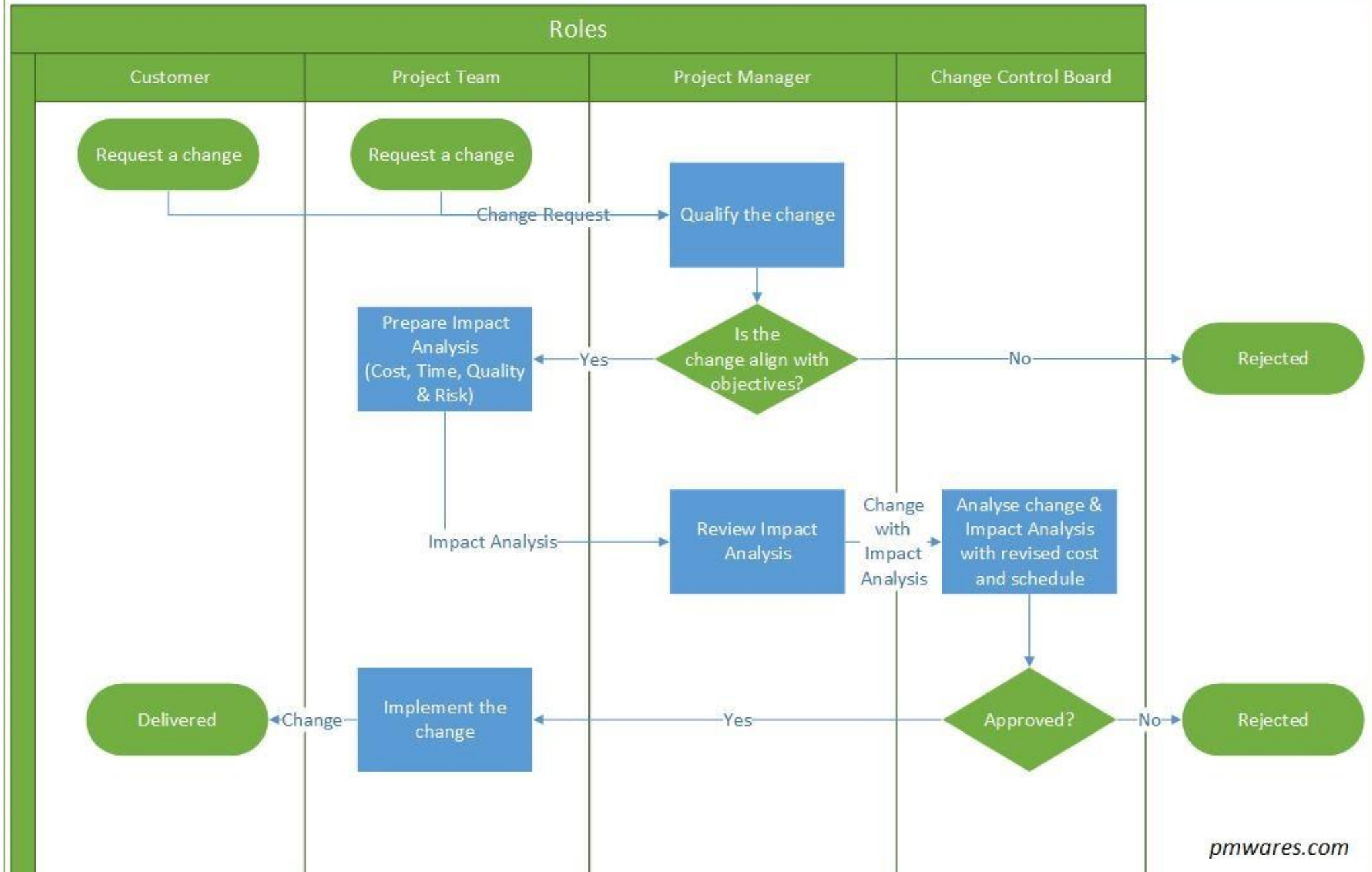
Change Control board

- Change requests are processed according to the change control system by the project manager, CCB, or by an assigned team member. Approved change requests will be implemented through the Direct and Manage Project Work process.
- The disposition of all change requests, approved or not, will be updated in the change log as part of updates to the project documents.
- A change log is used to document changes that occur during a project.
- These changes and their impact to the project in terms of time, cost, and risk, are communicated to the appropriate stakeholders. Rejected change requests are also captured in the change log

Change control process flows

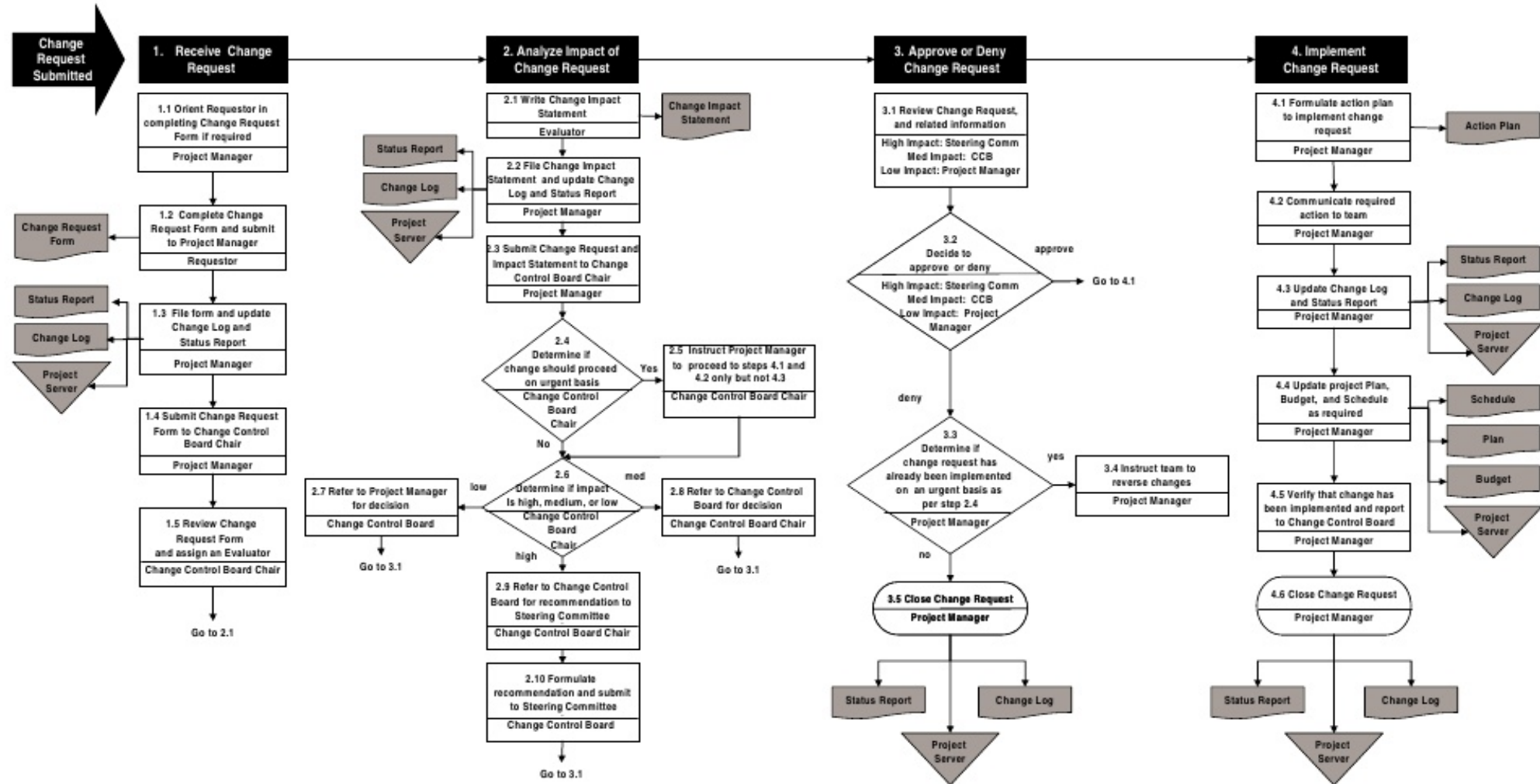


Change Management Process



6.3 More Complex Change Control Process Map

The following process map depicts a more complex manual change control process with a Project Manager, Change Control Board, and Steering Committee. Documentation includes a Change Log, Status Report, Change Request Form, and Change Impact Statement. There are eight types of change request status: submitted, analyzed, approved, denied, cancelled, implemented, verified, and closed.



Change Control Form

Section A

Project		Change Number
Controlled Item		Item Version
Identification of Aspect to be Change	For Document give section number / page number For Software give Module, Screen or Report name	
Change Details Include indication of importance and urgency		
Tick if Continued Overleaf <input type="checkbox"/>		
Requester of Change Print Name		Date Raised

Section B

Investigator of Change		
Impact, give details of other items affected		
Investigation Outcome Reject / Action at No Cost / Action at Cost	Suggested Priority High / Medium / Low	Date Investigated

Section C

<u>Implementor</u>		Date Scheduled
--------------------	--	----------------

Section D

Change Implemented <u>Implementator</u>	Signature	Date
Project Manager		

The background features a series of concentric circles in light gray, some solid and some dashed, creating a ripple effect. A large, solid red speech bubble is centered on the page, pointing downwards. The text "Risk Management Plan" is written in white, sans-serif font inside the speech bubble.

Risk Management Plan

What is risk management?

- Project Risk Management includes the processes of conducting risk management planning, identification, analysis, response planning, and controlling risk on a project.
- The objectives of project risk management are to increase the likelihood and impact of positive events, and decrease the likelihood and impact of negative events in the project.

What is a risk management plan?

- Risk Management Planning is the process of defining how to conduct risk management activities for a project.
- The key benefit of this process is it ensures that the degree, type, and visibility of risk management are commensurate with both the risks and the importance of the project to the organization.
- The risk management plan is vital to communicate with and obtain agreement and support from all stakeholders to ensure the risk management process is supported and performed effectively over the project life cycle.
- The key inputs to the Risk Management Plan are the project charter, the project management plan and Stakeholder Register

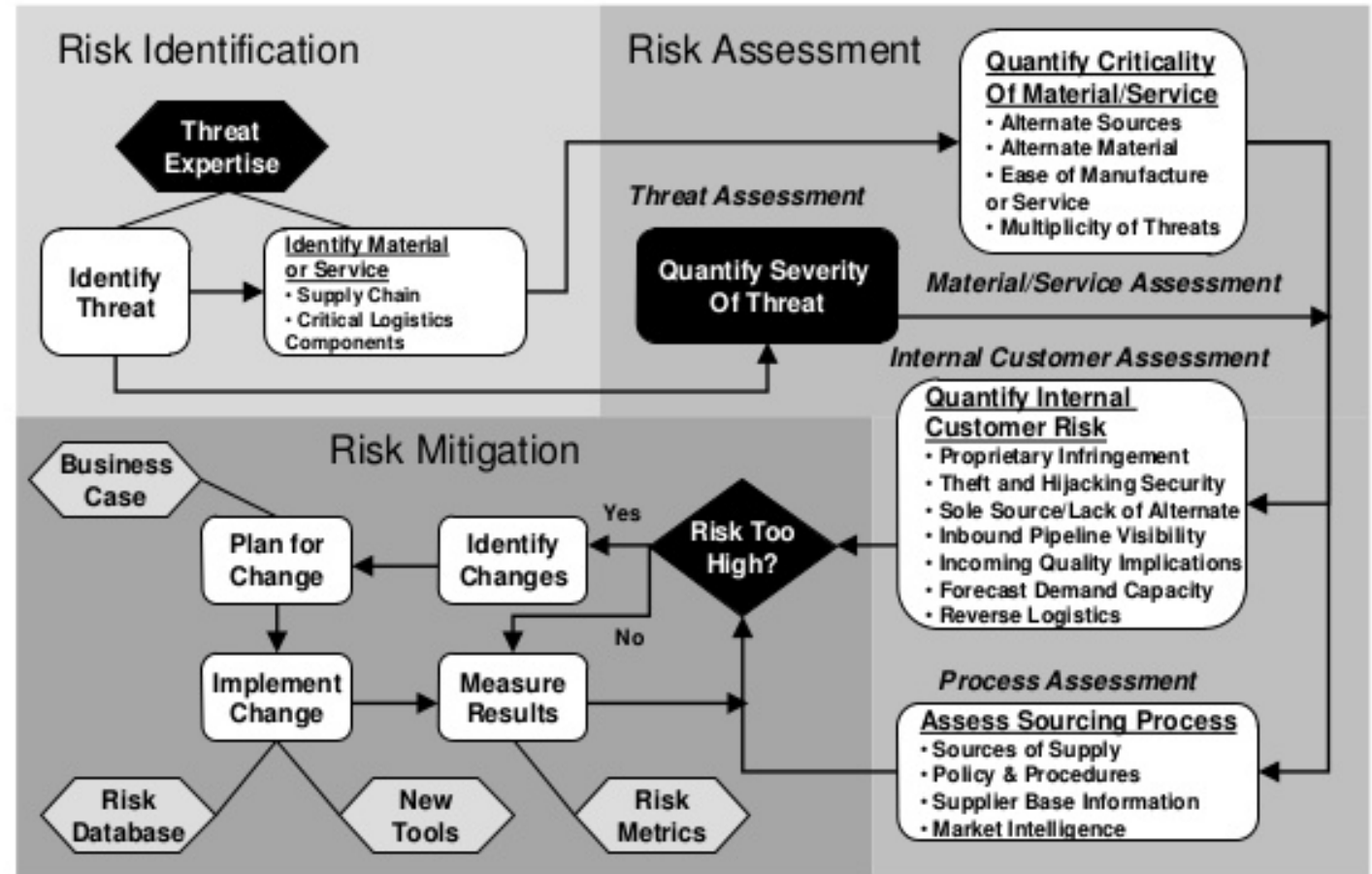
What is in a risk management plan?

The risk management plan is a component of the project management plan and describes how risk management activities will be structured and performed. The risk management plan includes the following:

- **Methodology** - Defines the approaches, tools, and data sources that will be used to perform risk management on the project.
- **Roles and responsibilities** - Defines the lead, support, and risk management team members for each type of activity in the risk management plan, and clarifies their responsibilities.
- **Budgeting** - Estimates funds needed, based on assigned resources, for inclusion in the cost baseline and establishes protocols for application of contingency and management reserves.
- **Timing** - Defines when and how often the risk management processes will be performed throughout the project life cycle, establishes protocols for application of schedule contingency reserves, and establishes risk management activities for inclusion in the project schedule.
- **Risk categories** - Provide a means for grouping potential causes of risk.
- **Definitions of risk probability and impact** - The quality and credibility of the risk analysis requires that different levels of risk probability and impact be defined that are specific to the project context.

Risk Management Flowchart—Example

Risk Management
Flowchart
Example



Risk Management Flow Sample

