

SDLC

Sujata Batra - sujatabatra@hotmail.com

- IT Trainer Since 2000
- More than 50+ Corporate Clients



S
u
j
a
t
a
B
a
t
r
a

Project :

Converting a vision, a dream or a need to reality.

- ✓ A job that has a beginning and an end (Time)
- ✓ A specified outcome(Scope)
- ✓ At a stated level of Performance (Quality)
- ✓ At a budget(Costs).



Project Characteristics :

- *Temporary* : Has definite Start and Finish
- *Unique* : Product/Service is different in some distinguishing way

What is Project



- Project, as defined in the PMBOK Guide (Project Management Body of Knowledge), is a “temporary endeavor undertaken to create a unique product, service or result.”
- All projects vary in complexity but they all follow similar life cycles
- All projects have deliverables (meaning they always produce something)
- Project examples:
 - New product development
 - Building renovation
 - Wedding
 - Dinner party

Management :

Management is the technique of understanding the problems, needs and controlling the use of Resources, Cost, Time, Scope and Quality.

Project Management :

Application of **knowledge, skills , tools & techniques** to project activities in order to meet **stakeholder** needs & expectations from a project.

Needs : stated part of the project

Expectations : unstated part of the project

“Completion of Project on time within Budget without comprising Quality”

Why do people learn PM?

- To explore the latest concepts and techniques of project management.
- To increase value/contribution to the organization. To prove yourself skillful in managing projects.
- To learn a new thought process that helps organized thinking and structured approach.
- To acquire a professional degree/ recognition and increase job prospects.
- Endless possibilities and benefits.....

Project Management vs Operation Management:

- *Project management* is the process and activity of planning, organizing, motivating, and controlling resources, procedures and protocols to achieve specific goals in scientific or daily problems.
- *Operations management* is an area of management concerned with overseeing, designing, and controlling the process of production and redesigning business operations in the production of goods or services.

Project Management Vs Operation Management:

Basis Of:	Project Management	Operations Management
Time	Projects have a definitive start and a definitive finish.	Operations are continuous.
Task Type	Projects tasks are specific for that for that project and have never been done before.	Operational tasks are repetitive and cyclical.
Success/Improvement Criteria	Project work success is based on project objectives identified specifically and uniquely for that project	Operation Work success is based on previous indicator(i.e system availability)
Change Type	Projects implements revolutionary change	Operations implements evolutionary change
Team continuity	Project Teams are formed to implement project and then disbanded once project is completed	Stable Organization/ Operational Teams are consistent

S
u
j
a
t
a
B
a
t
r
a

Project Management Vs Operation Management:

Basis Of:	Project Management	Operations Management
Team Composition	Project Team consist of team members from different departments , different skill sets.	Operational teams frequently consist of team members with similar technical skill sets.
Governance	Project Manager has accountability and authority <ul style="list-style-type: none">• Stakeholders• The business• Operations• Projects can have a flat hierarchy Career paths for team members is not clear• PM sets the priorities.	Organizational Structure <ul style="list-style-type: none">• Multi level - Hierarchy• Clearly defined career paths• Needs must defines the priorities (System down is a priority).
Budget Estimation	Difficult to estimate time and budget	Budget set and fixed events

Project Management Plan :

“Tells How work will be done”

The key to a successful project is on the planning. All the detailed planning work for different aspects of the project is integrated into one single plan known as the Project Management Plan.

Input:

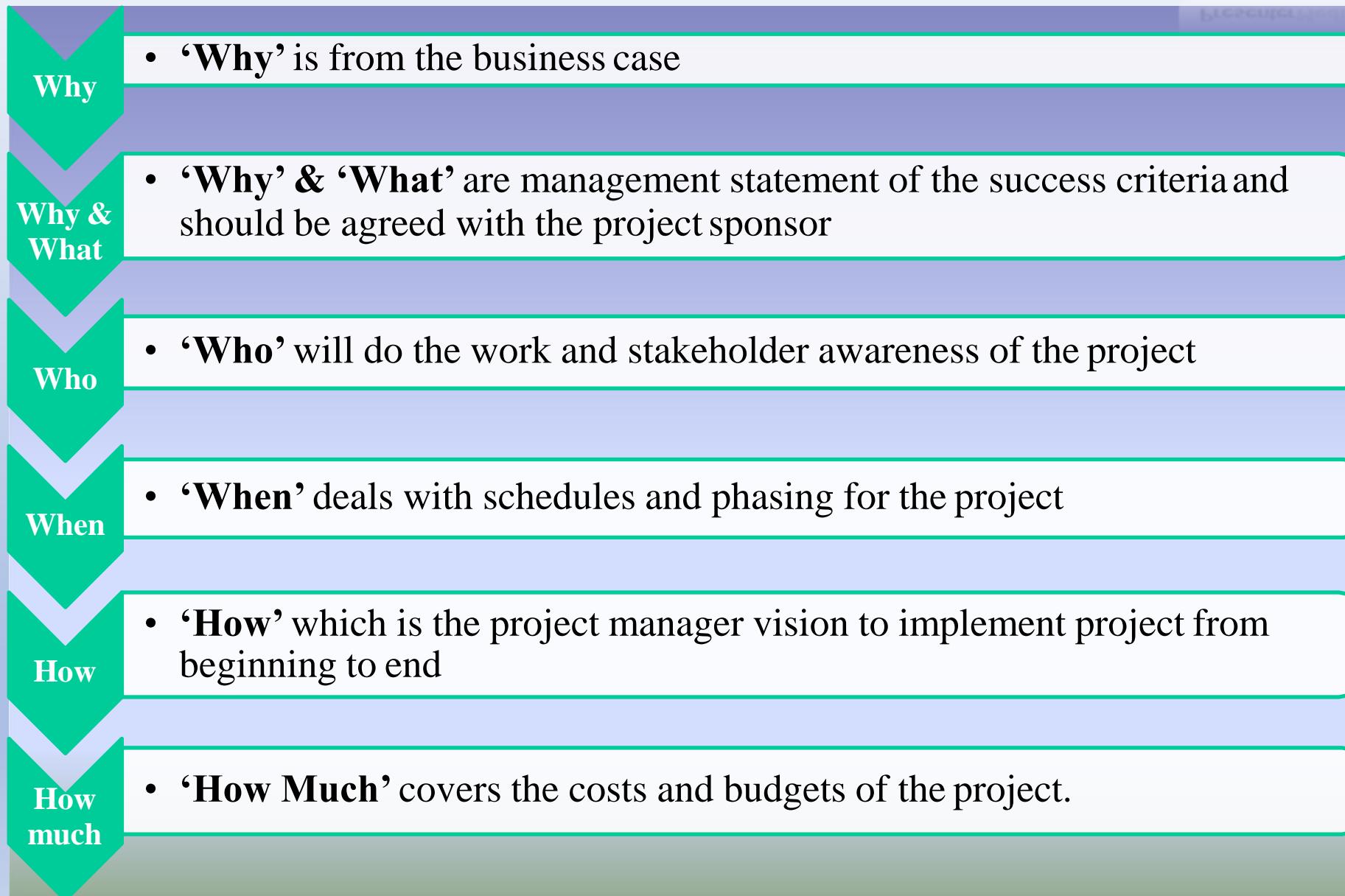
Goal
Team
Money
Time
Equipment

Project

Output :

Deliverables

The PM Plan establishes the projects:



S
u
j
a
t
a
B
a
t
r
a

Project Management Overview

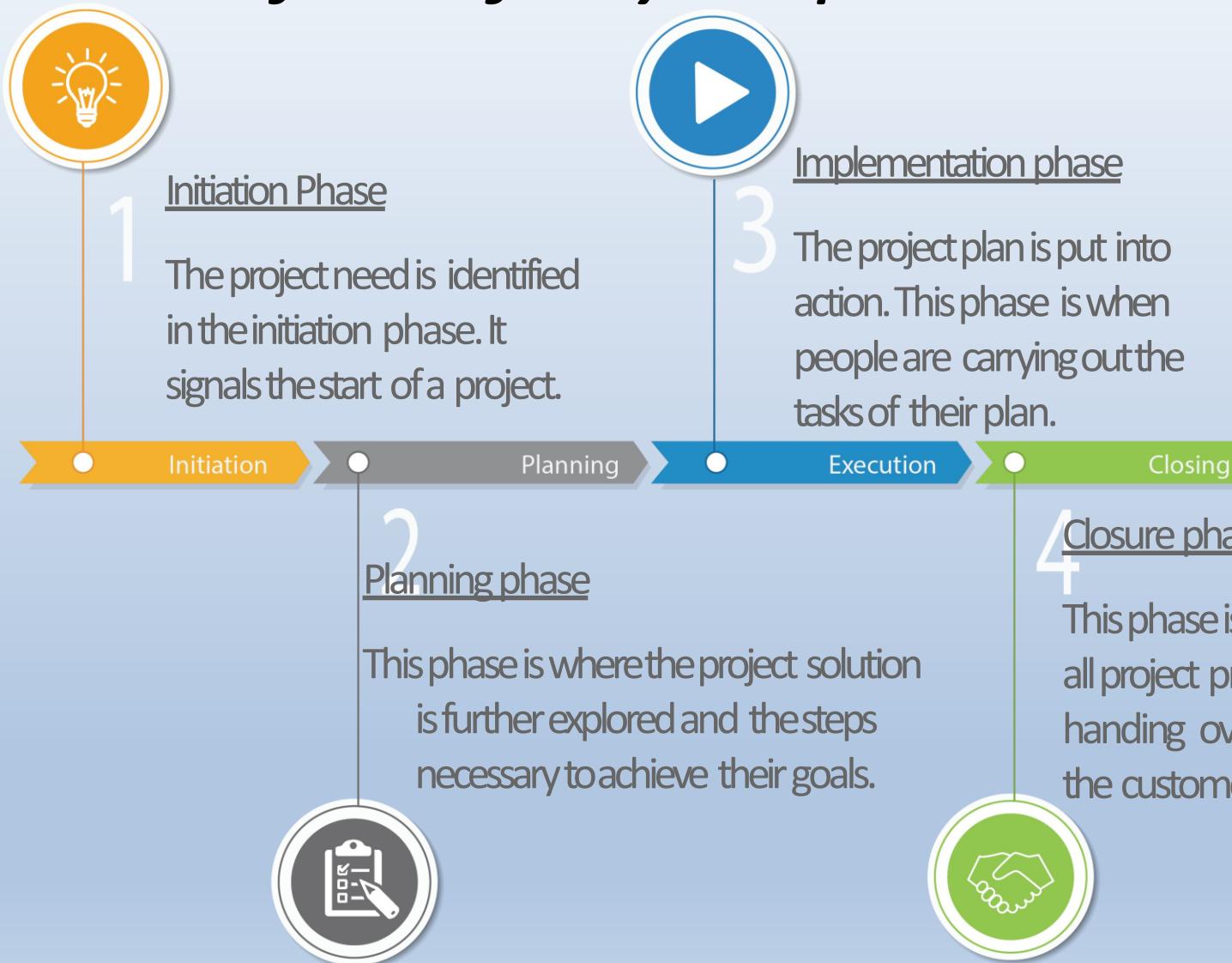
Project - Current Challenges



What is Project Life Cycle?

- Project life cycle is a series of phases of a project from initiation to completion.
- The life cycle gives a practical approach to problem solving applied to all aspects of a project.
- Phases in a project life cycle encompasses sequential and overlapping phases.
- A project life cycle typically has 4 major phases:
 - Initiation Phase
 - Planning Phase
 - Implementation/ Execution Phase
 - Closure Phase

Project life cycle phases



Initiation Phase

- The first phase explores the project concept. Scope is defined during this phase.
- Feasibility studies are made in order to identify if there is a business need and justification to pursue the project. Project charter is developed for approval.
- This is the phase that the project team is assembled and the project manager is identified.

Planning Phase

- This phase further details the scope of the project.
- Tasks and resources are identified and assigned during this phase
- Project manager coordinates the preparation of the schedule and project budget
- Risks are identified ahead to anticipate any project threats
- Quality plan is developed to maintain proper standards throughout project
- Communications plan is created in order to ensure everyone is constantly informed of project status

Implementation Phase

- This phase is where the work outlined in the project plan is performed.
- This phase consumes the most resources and energy.
- Constant and close monitoring of the work should be done to ensure efficiency of the project execution.
- Status reports are important for all stakeholders involved.
- Deliverables are measured against the set metrics to ensure quality is acceptable

Closure Phase

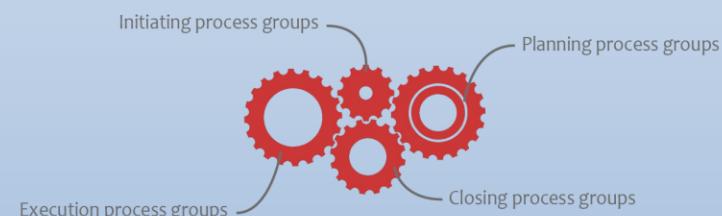
- This is the last phase of the project life cycle and involves handing over final deliverable to the customer.
- Contracts are properly terminated for equipment, vendors and staff.
- All stakeholders are to be informed of project closure.
- This phase is when the team reviews the overall project and identify lessons learned for future projects.

Processes



Processes are the tasks and methodologies that occur within a phase.

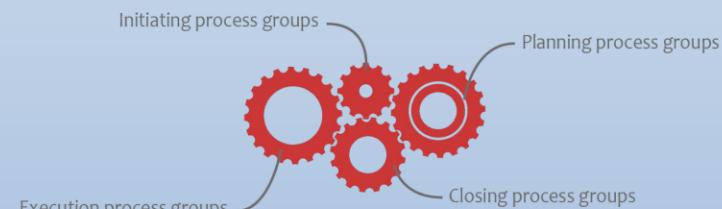
Although they sound like phases, processes are not phases.



Processes

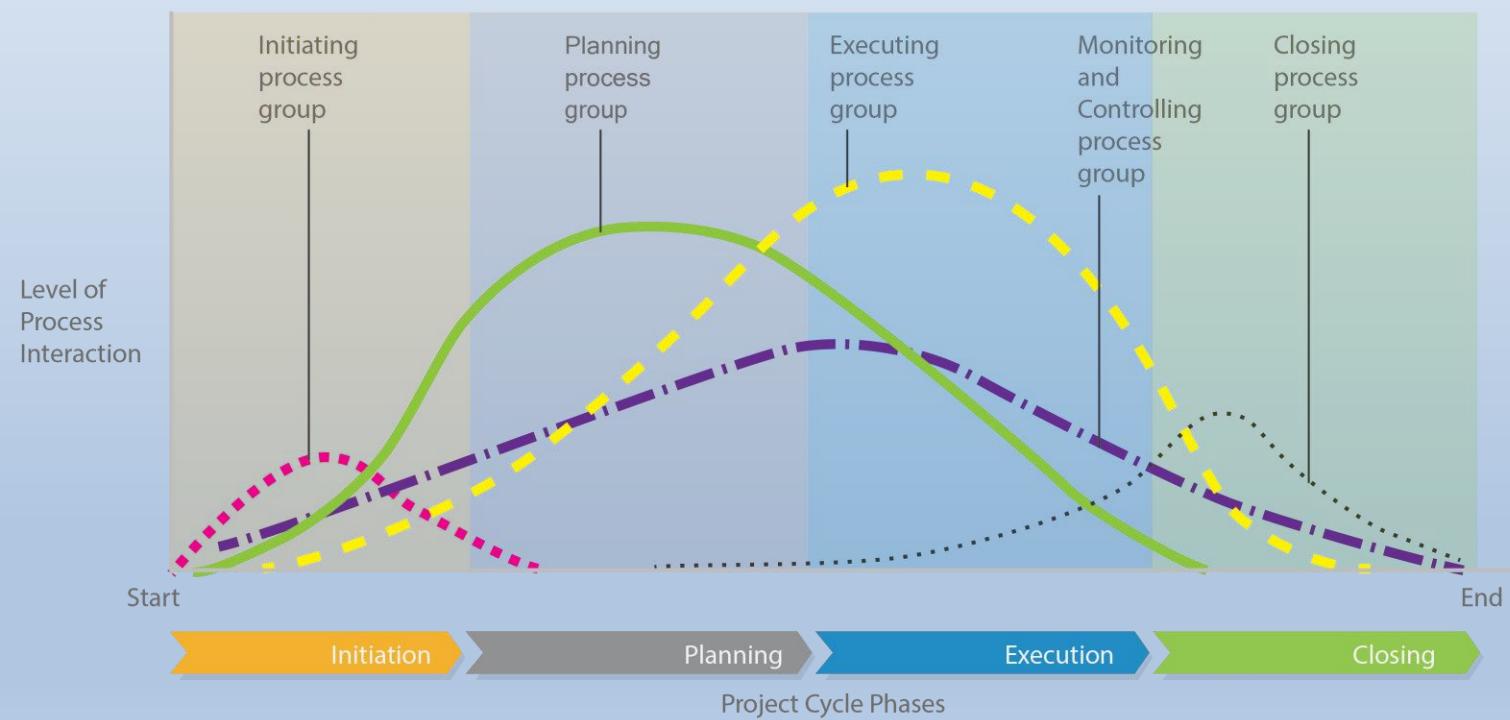


The PMBOK arrange processes into groups according to when they typically occur in a project.
However, different processes can occur any time during a project phase.

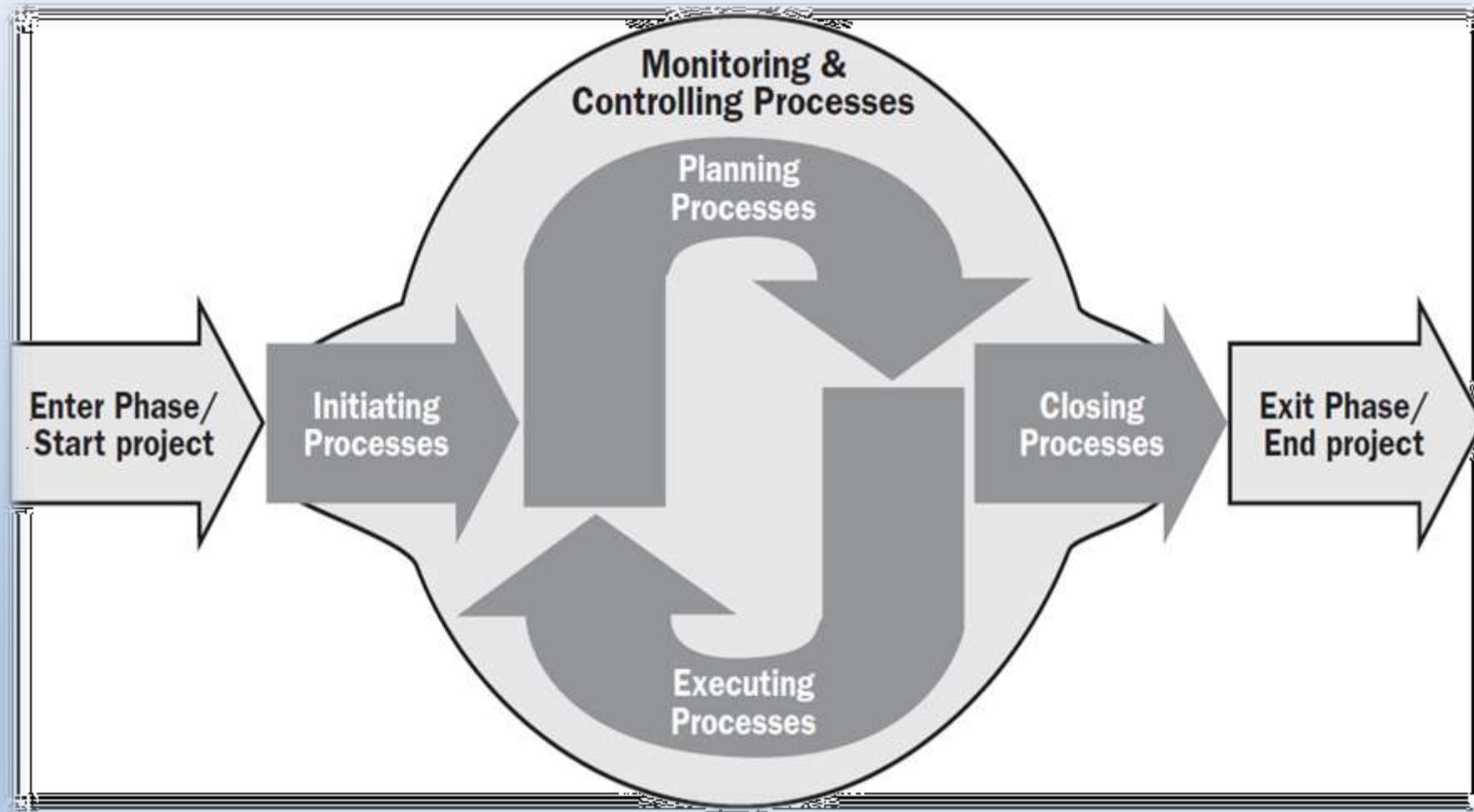


Processes

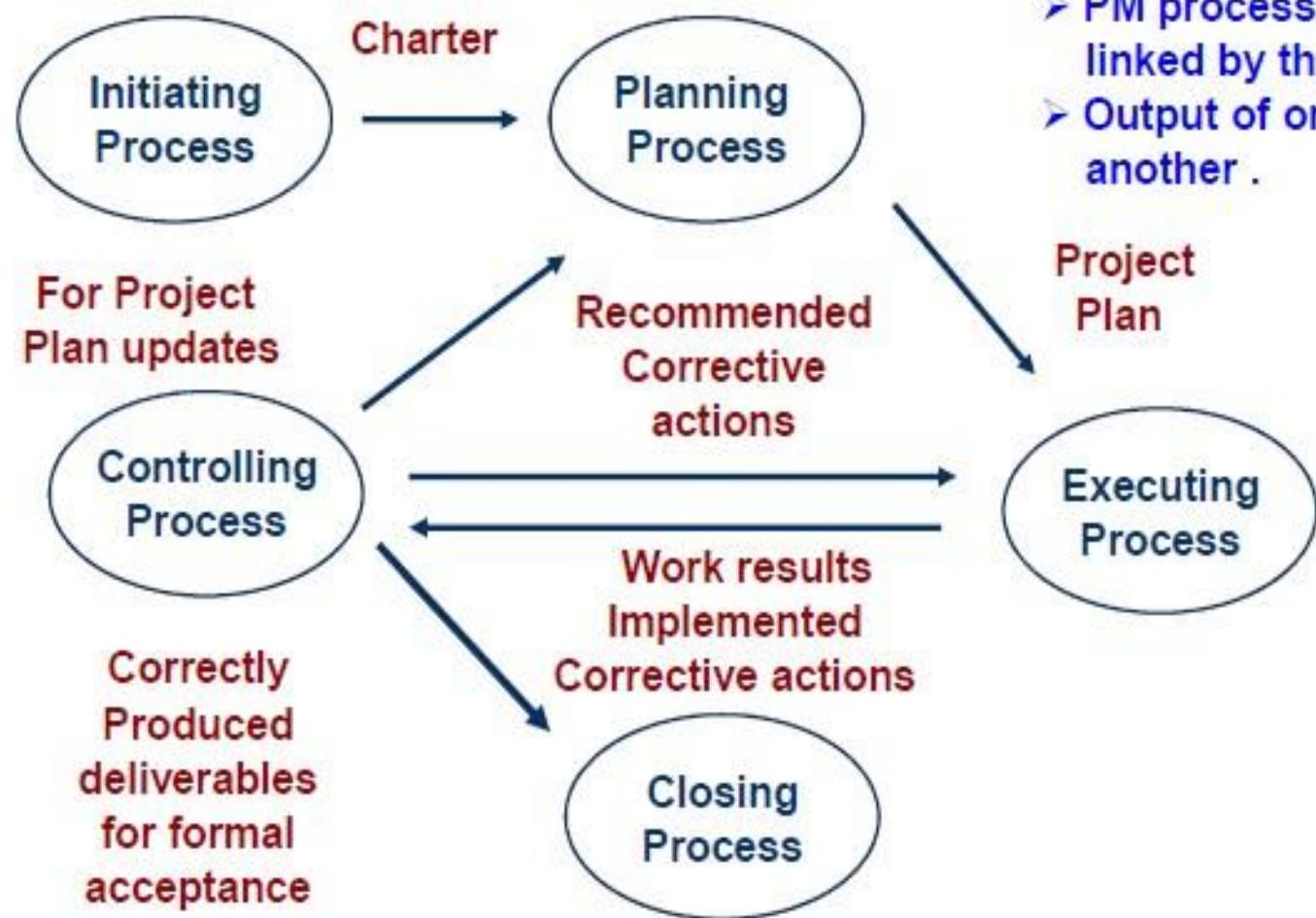
- This graphic is a representation of when these processes typically occur over the project life cycle



Project Management Process



Project Management Process Groups



- PM process groups are linked by their outputs.
- Output of one is input to another .





Initiating process group

- **Typically occurring during the initiation phase of a project.**
- **These process groups define the new project or phase objectives and what is needed to achieve it.**
- **Involves obtaining authorization for the project or phase.**
- **Process group main output is the project charter and stakeholder register.**

Initiation processes:

4.1 Develop Project Charter

13.1 Identify Stakeholders



From PMBOK, 5th edition





Planning process group

- ② **Typically occurring during the planning phase of a project.**
- ② **Planning process group does not just apply to the planning phase. Planning is an activity that continues to almost the end of a project, as projects tend to change as it progresses.**
- ② **A skilled project manager knows how much time and effort to put into each stage of planning as the project changes**

Planning processes:

- 4.2 Develop Project Management Plan
- 1. Plan Scope Management
- 2. Collect Requirements
- 3. Define Scope
- 4. Create WBS
- 5. Plan Schedule Management
- 6. Define Activities
- 7. Sequence Activities
- 8. Estimate Activity Resources
- 9. Estimate Activity Duration
- 10. Develop Schedule
- 11. Plan Cost Management
- 12. Estimate Costs
- 13. Determine Budget
- 8.1 Plan Quality Management
- 9.1 Plan Human Resource Management
- 10.1 Plan Communications Management
- 1. Plan Risk Management
- 2. Identify Risks
- 3. Perform Qualitative Risk Analysis
- 4. Perform Quantitative Risk Analysis
- 5. Plan Risk Responses
- 12.1 Plan Procurement Management
- 13.2 Plan Stakeholder Management

From PMBOK, 5th edition





Executing process group

- **Typically occurring during the implementation/execution phase of a project.**
- **This process group involves a lot of management tasks and coordination with staff while the project plan is being executed.**
- **The outputs produced during this process group is the actual project deliverable.**

Execution processes:

- 4.3 Direct and Manage Project Work
- 8.2 Perform Quality Assurance
- 2. Acquire Project Team
- 3. Develop Project Team
- 4. Manage Project Team
- 10.2 Manage Communications
- 12.2 Conduct Procurements
- 13.3 Manage Stakeholder Engagement



From PMBOK, 5th edition





Closing process group

- **Typically occurring during the closure phase of a project.**
- **Formal transfer of deliverable to customer**
- **Administrative closure and contract closure is performed to formally close out project.**

Closing processes:

4.6 Close Project or Phase

12.4 Close Procurements



From PMBOK, 5th edition





Monitoring and controlling process group

- **This process group occurs on all phases of the project life cycle.**
- **Ensures that the project is on track and flags issues to maintain project course direction.**
- **Collects performance data through constant reports from team such as status reports and team meetings.**

Monitoring and Controlling processes:

4. Monitor and Control Project Work
5. Perform Integrated ChangeControl
5. Validate Scope
6. Control Scope
- 6.7 Control Schedule
- 7.4 Control Costs
- 8.3 Control Quality
- 10.3 Control Communications
- 11.6 Control Risks
- 12.3 Control Procurements
- 13.4 Control Stakeholder Engagement

From PMBOK, 5th edition



Final thoughts

- While the PMBOK and PMI gives an outline of a typical project, no two projects are the same.
- A good project manager knows how to choose which processes are needed for their project.
- Practicality and pragmatism is the best approach to any project. (keep it simple)



PM Knowledge Areas

S
u
j
a
t
a
B
a
t
r
a



Project Integration Management

❖ **Project Integration Management** supports various elements of project management which are identified, defined, combined and coordinated.

- Develop Project Charter
- Develop Project Management Plan
- Direct and Manage Project execution
- Monitor and Control Project Work
- Perform Integrate Change Control
- Close Project or Close



Project Integration Management

Project Charter :

- ✓ DOCUMENT that formally authorizes a project.
- ✓ Provide information about internal and external parties involved in and affected by the project.
- ✓ Documenting initial requirements that satisfy the stake holders' needs and expectations.
- ✓ ISSUED by a project initiator or sponsor, external to project organization, at a level appropriate to project funding.
- ✓ EMPOWERS the project manager to apply resources to project.
- ✓ Summary level Milestone schedule and Summary level Budget.



S
u
j
a
t
a
B
a
t
r
a

Project 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

- Collect requirements
- Define Scope
- Create WBS
- Verify Scope



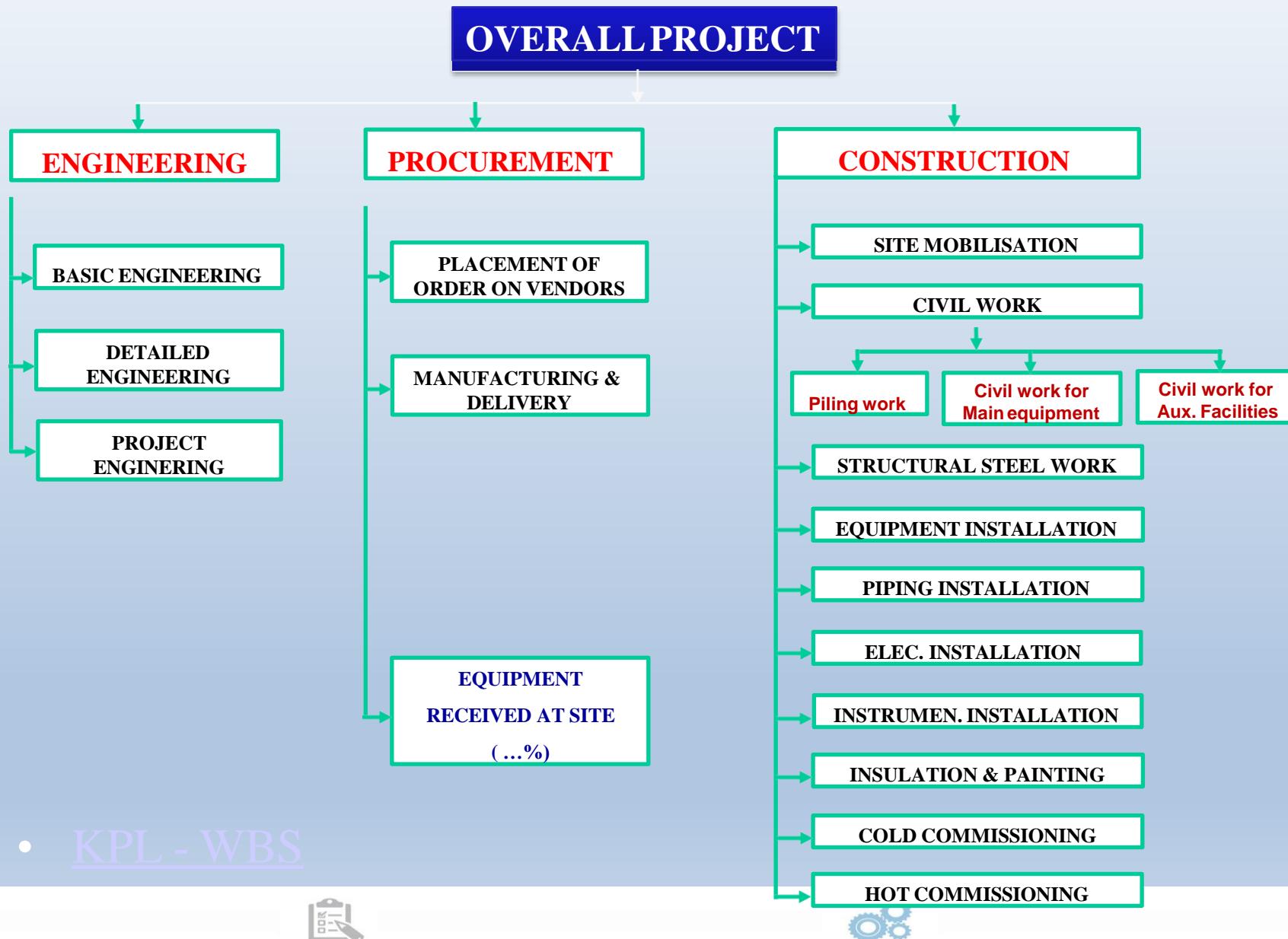
Project Scope Management

WBS – Work Breakdown Structure :

- ✓ A deliverable-oriented hierarchical decomposition of work to be executed by the project team to:
 - create required deliverables
 - accomplish project objectives
- ✓ WBS organizes and defines the total scope and represents specified in the current approved Scope Statement!
- ✓ Process of subdividing project deliverables and project work into more manageable components.
- ✓ Lowest level of WBS is work package can be scheduled, cost



TYPICAL WBS OF A PROJECT



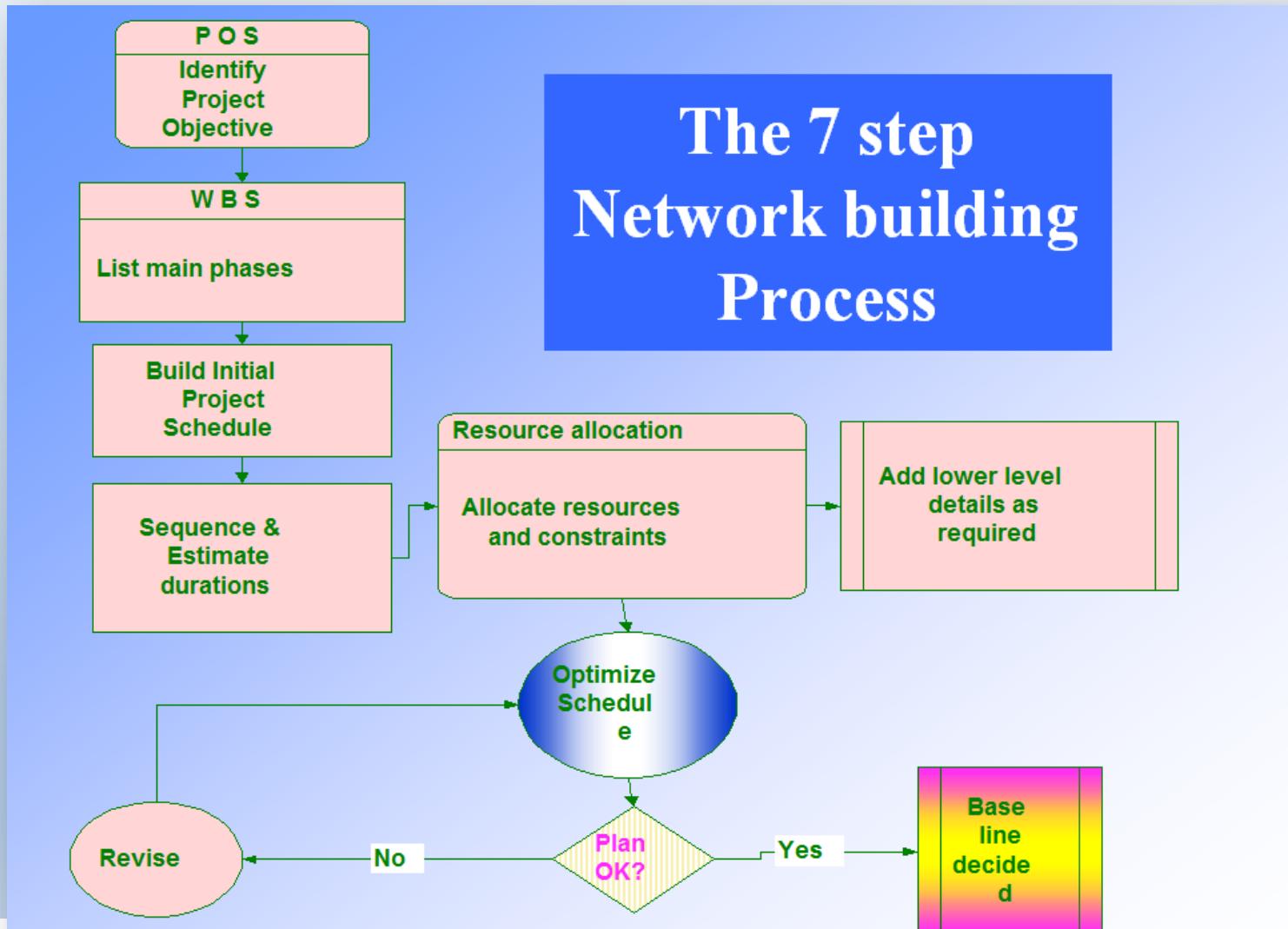
Project Time Management

❖ Project Time Management ensures the timely completion of the project.

- Plan Schedule Management
- Define Activities
- Sequence Activities
- Estimate Activity Resources
- Estimate Activity Durations
- Develop Schedule
- Control Schedule



Project Time Management



PresenterMedia



Project Time Management

- ✓ Schedule once finalized is set as a baseline
- ✓ Progress of work are tracked against the baseline
- ✓ Current progress is arrived from various (Engineering, Proc, Manufacturing and Construction) Trackers developed for the purpose. These schedules are called current schedules
- ✓ Analysis and forecasting is done in progress reports.



PresenterMedia



Project Cost Management

❖ Project cost Management includes the processes involved in estimating, budgeting, and controlling costs so that the project can be completed within the approved budget.

- Estimate cost
- Convert it into budget
- Load the cost into schedule
- Perform earned value (EV) analysis
- Perform estimate at complete (EAC)
- Administer changes
- Control cost



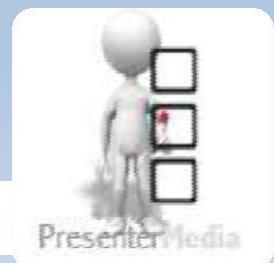
Project Quality Management

- ❖ Project Quality Management ensures the project will satisfy NEEDS for which it was undertaken.

- Plan Quality
- Perform Quality Assurance
- Perform Quality control

What is Quality?

- ✓ Conformance to specifications
- ✓ Fitness for use



Project Quality Management

Plan Quality Techniques

- Cost benefit analysis
- Cost of quality
- Benchmarking
- Design of experiments
- Seven quality tools
- Statistical sampling

Perform Quality Assurance Techniques

- Quality management and control tools
- Quality audits
- Process analysis

Perform Quality Control Techniques

- Statistical sampling
- Inspection
- Seven quality tools and techniques
 - ✓ Cause and effect diagram
 - ✓ Flowcharts
 - ✓ Check sheets
 - ✓ Pareto diagrams
 - ✓ Histogram
 - ✓ Control charts
 - ✓ Scatter diagram



Project Human Resource Management

❖ Project human resource management includes the processes that organize, manage, and lead the project team and to make most effective use of people involved in the project.

- ❑ Develop Human Resource Plan (Role & Responsibility, Organization chart, Staffing Management Plan)
- ❑ Acquire project team
- ❑ Develop project team
- ❑ Manage project team



Presenter Media



Organization Chart

Project Human Resource Management

Project HR Management



Project Communications Management

❖ Project communication management include processes required to ensure timely and appropriate generation, collection, distribution, storage, retrieval, and ultimate disposal of project information.

- Identify Stakeholders
- Plan Communication
- Distribute Information
- Manage Stakeholders expectations
- Report Performance

RACI



PresenterMedia



Project Communications Management

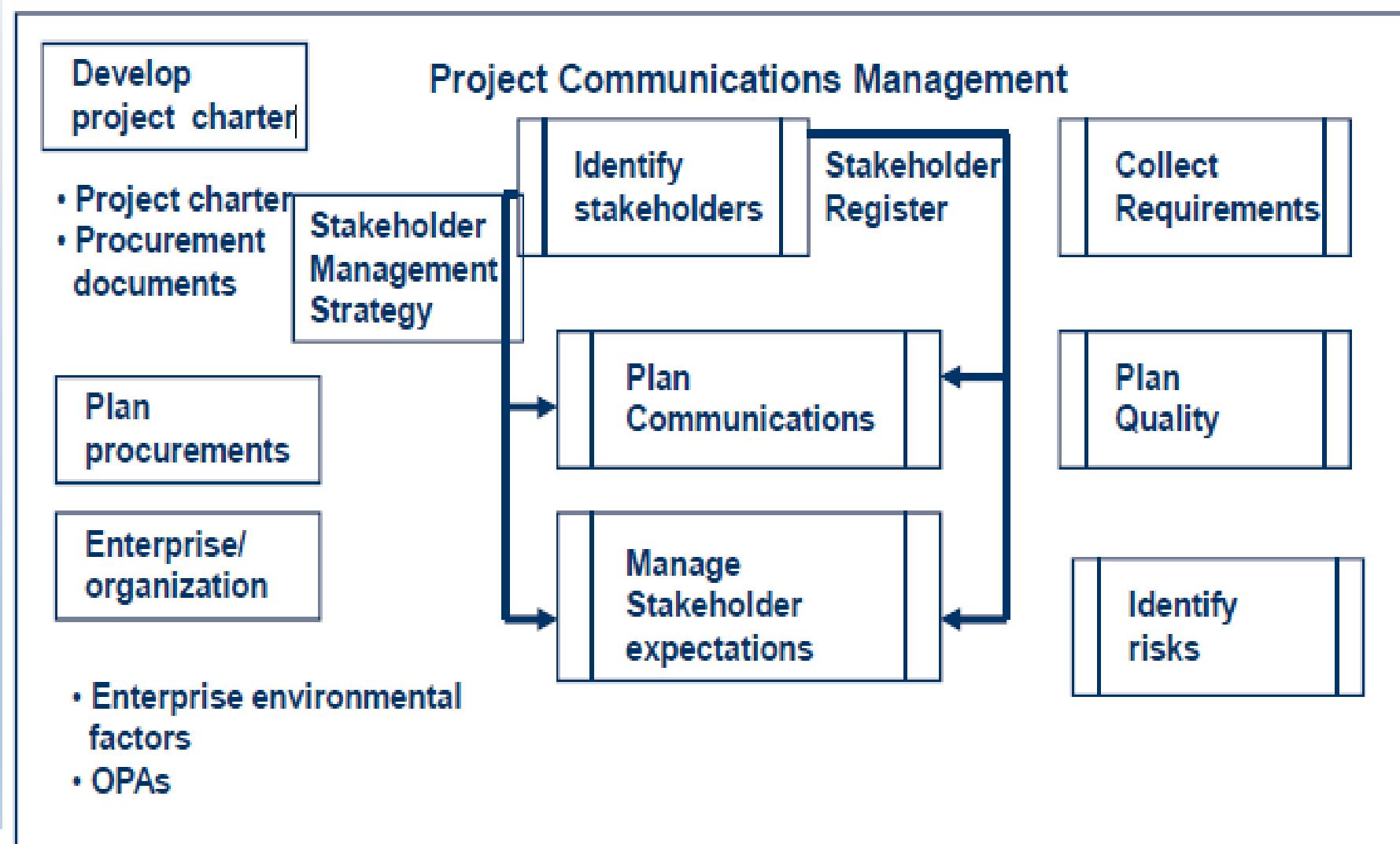
S
u
j
a
t
a
B
a
t
r
a

Project Stakeholders :

- ✓ Customers/users
- ✓ Sponsor
- ✓ Portfolio managers/portfolio review board
- ✓ Program managers
- ✓ Project management office
- ✓ Project managers
- ✓ Project team
- ✓ Functional managers
- ✓ Operations management
- ✓ Sellers/business partners



Project Communications Management



Project Risk Management

- ❖ Project Risk Management is concerned with identifying, analyzing and responding to project risks.

- Plan Risk Management
- Identify Risks
- Perform Qualitative Risk Analysis
- Perform Quantitative Risk Analysis
- Plan Risk responses
- Monitor and Control Risks

Risk Register

Risk
Assessment

Risk
Control

Risk
Ranking

Risk
Mitigation



Project Procurement Management

❖ Project Procurement Management is needed to acquire material, goods and services outside performing organization to meet project scope.

- Plan Procurements
- Conduct Procurement
- Administer Contracts
- Close Procurement



Project Change Management

- ❖ Projects generally don't go 100% as planned resulting into variations from plans in scope, time, cost, quality
- ❖ Change control boards (CCBs)



Project Document Management

Document management is very important (but largely ignored) aspect of project management

- ❖ This can be a sophisticated software or simple excel sheets depending on the organization strategy
- ❖ Once adopted its use should be fully exploited viz. 100% documents should pass through it and document reviews should be done as per the matrix defined
- ❖ MIS of document status be generated at the end of period



Project Document Management

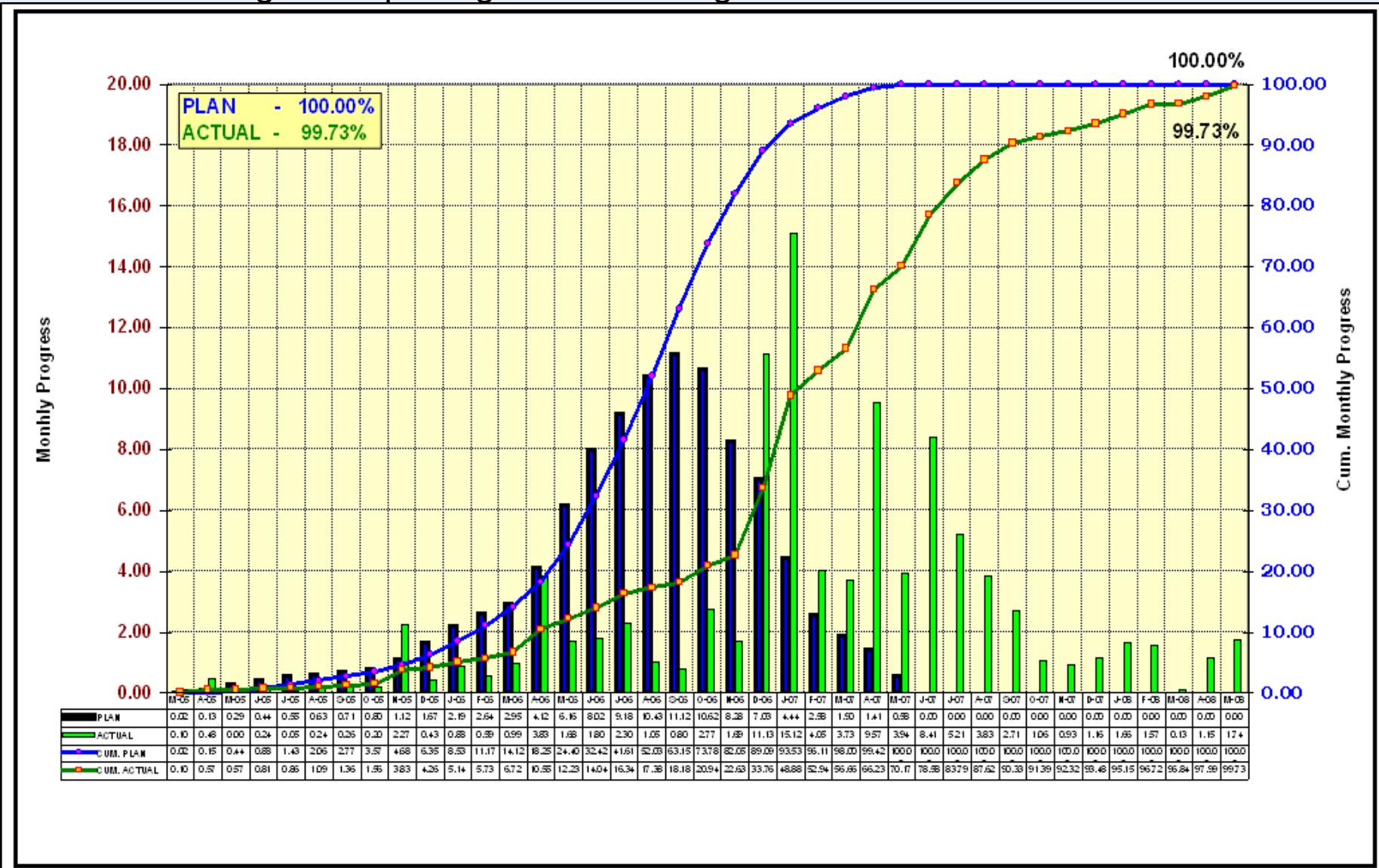
- ❖ Proper DMS ensures right information to all stakeholders at all times
- ❖ Avoid using obsolete drawings and designs for execution
- ❖ Serves a very good engine for future use of project data
- ❖ Example of a simple document management system

Document control log



S-Curve in Monthly Progress Report

Progress reporting is done through different formats and curves.



What is "S Curve"

S- Curve is the graphic display of cumulative progress plotted against time.

Ideal S - Curve is a sinusoidal curve based on the following formulae:

$$Y = [1 - \sin(x/x_n * 180 + 90) * 50]$$

Y – Percent progress

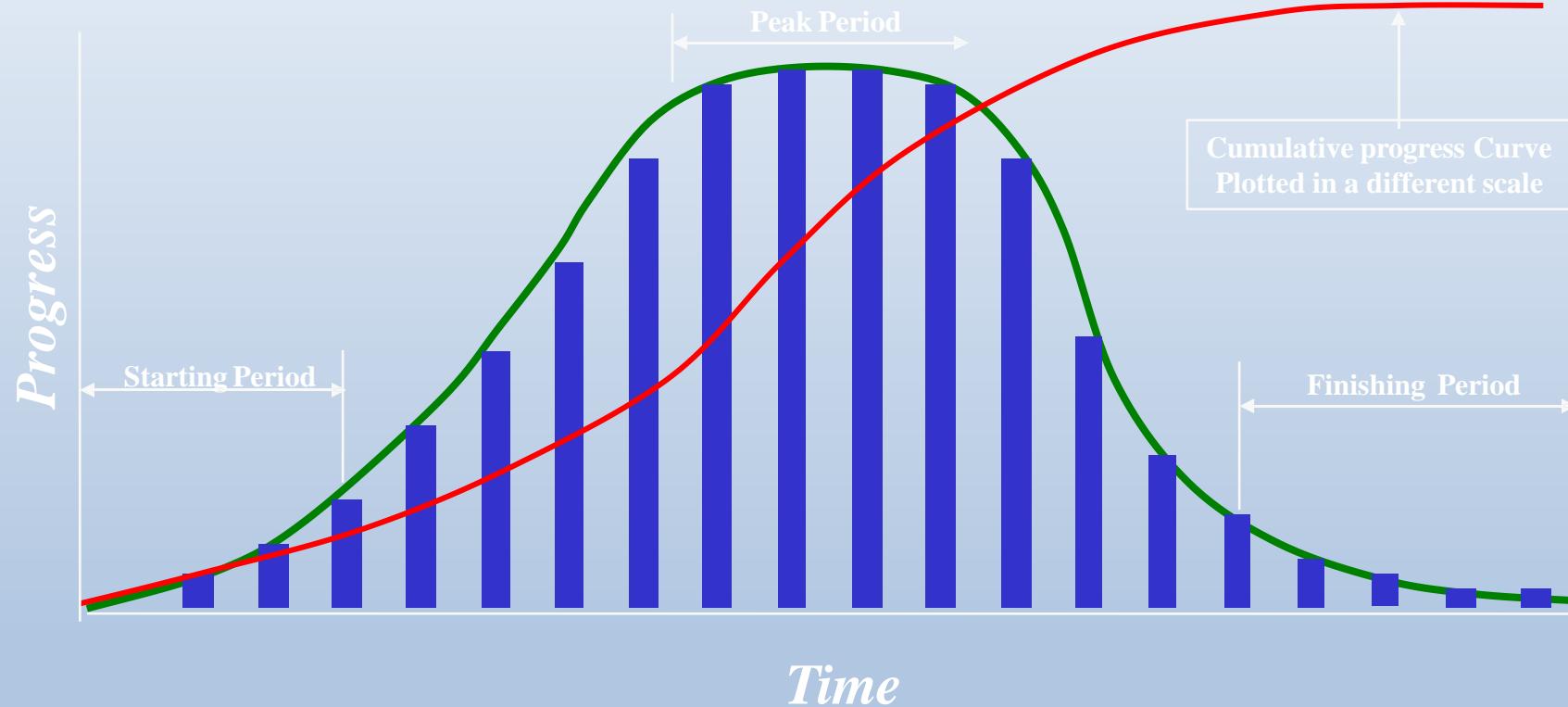
x – Period at which s-curve value required x_n –
Total period

The name is derived from the 'S' like nature of the curve.

S-Curve



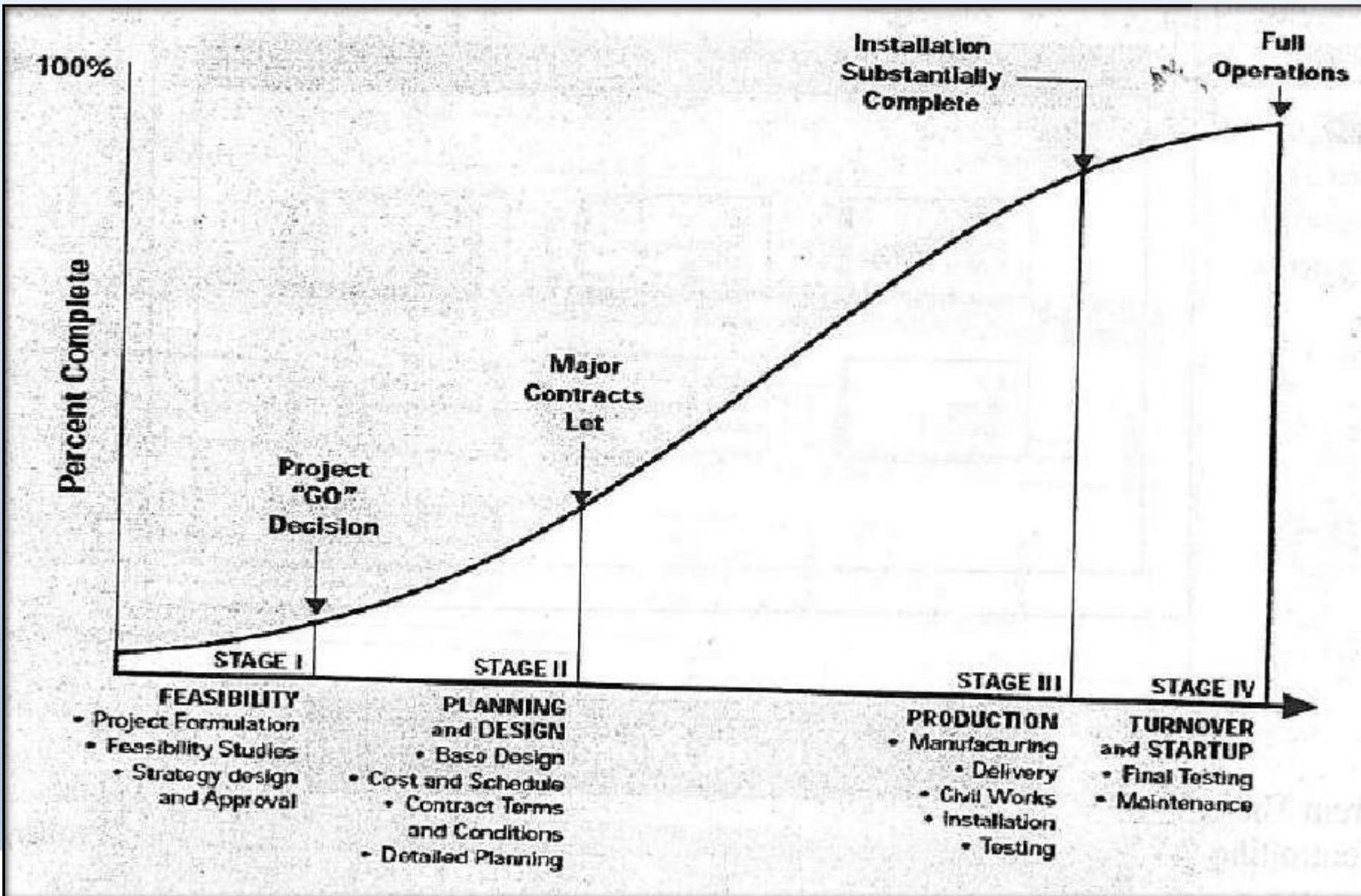
Progress distribution curve



It is an ideal distribution curve. Depending upon the various guiding factors it may vary.



"S-Curve" or Project Life Cycle



Key Competencies and Knowledge Areas in BA



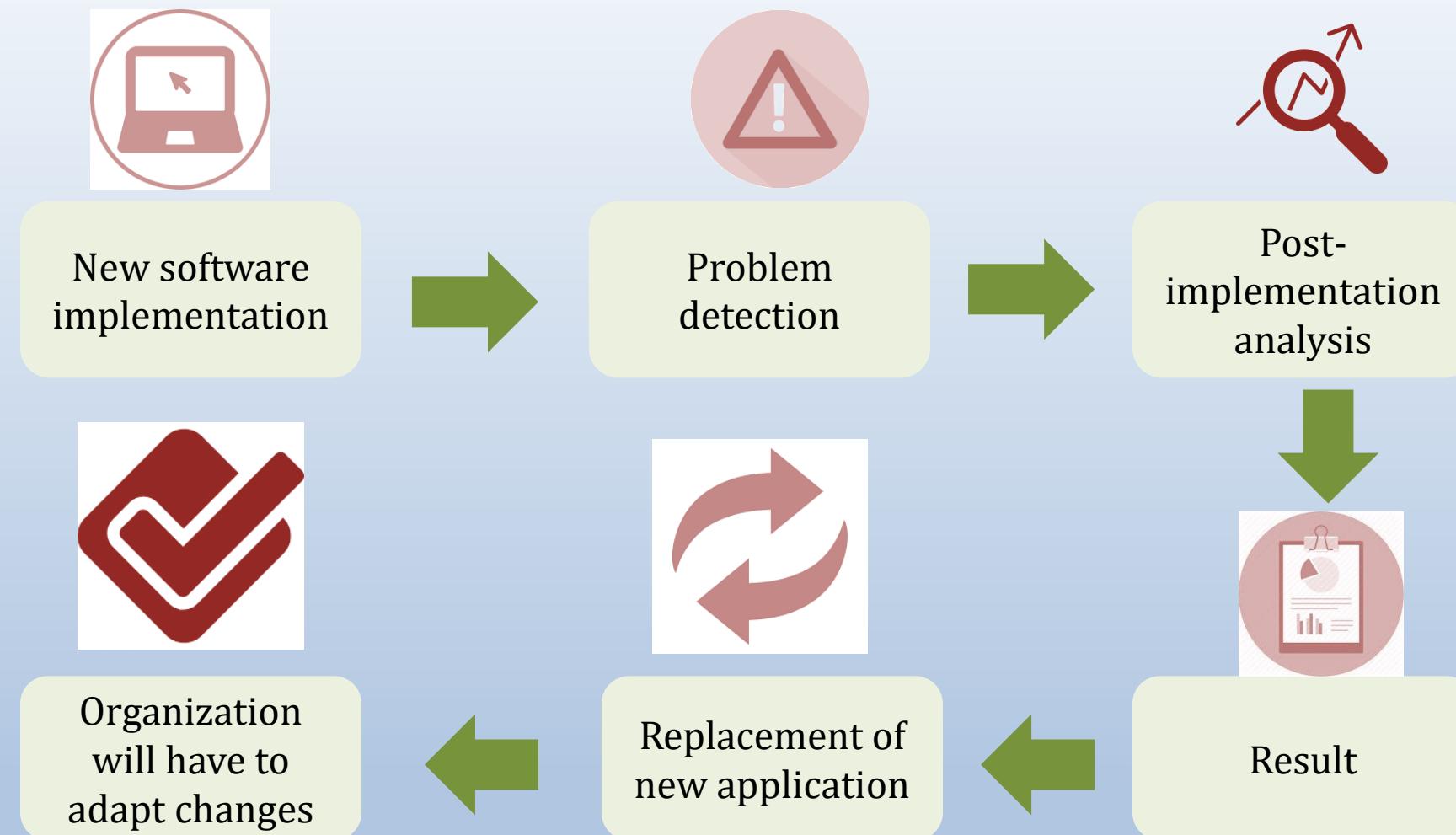
Champion of Change - The Business Analyst



Example: A newspaper printing press as change ... where technology is the problem and technology is the solution.



Champion of Change – Example



Project success doesn't just mean delivering on time and on budget but includes genuine acceptance by the people in the organization who will benefit or, in some cases, be impacted negatively by the new implementation.



S
u
j
a
t
a
B
a
t
r
a

Champion of Change - The Business Analyst

This aspect of the project is often lightly touched upon or brushed aside as the BA views these changes as part of :

01



Project
Communications



"Just tell the employees what
needs to be done
in the future"

01



Training



"Just send them to a training
session".

In reality, acceptance of change can either make or break a project.

The implementation of the application will always result in some change to the organization's structure, processes, systems and/or jobs.



S
u
j
a
t
a
B
a
t
r
a

Champion of Change - The Business Analyst

Whenever change of any type happens within the organization it requires extensive planning and hard work to ensure the change is implemented in the most effective manner.

This change applies to everyone involved in the project:



Individuals



Stakeholders

Mark Twain's quotes,

"I'm all for progress. It is change I don't like."



S
u
j
a
t
a
B
a
t
r
a

Champion of Change - The Business Analyst

S
u
j
a
t
a
B
a
t
r
a

1

The first step to overcome change is to identify key influencers/stakeholders. Key influencers are involved from the start of the project.

2

These key influencers are shown a working prototype.

3

Then, these stakeholders are trained to get their consent. Convincing others in the team is left to the key influencers.



IIBA and BABOK

International Institute of Business Analysis™ (IIBA®) is the recognized, non-profit association for business analysis professionals around the world.



International Institute
of Business Analysis

As the voice of the business analysis community, IIBA supports the recognition of the profession and works to maintain standards for the practice and certification.



Knowledge Areas in the IIBA®'s BABOK® Guide

The tasks followed by business analyst require specific knowledge areas and a business analyst should focus on them while performing the activities.



BABOK (Business Analysis Body Of Knowledge) has also defined these knowledge areas which group together related tasks and techniques.

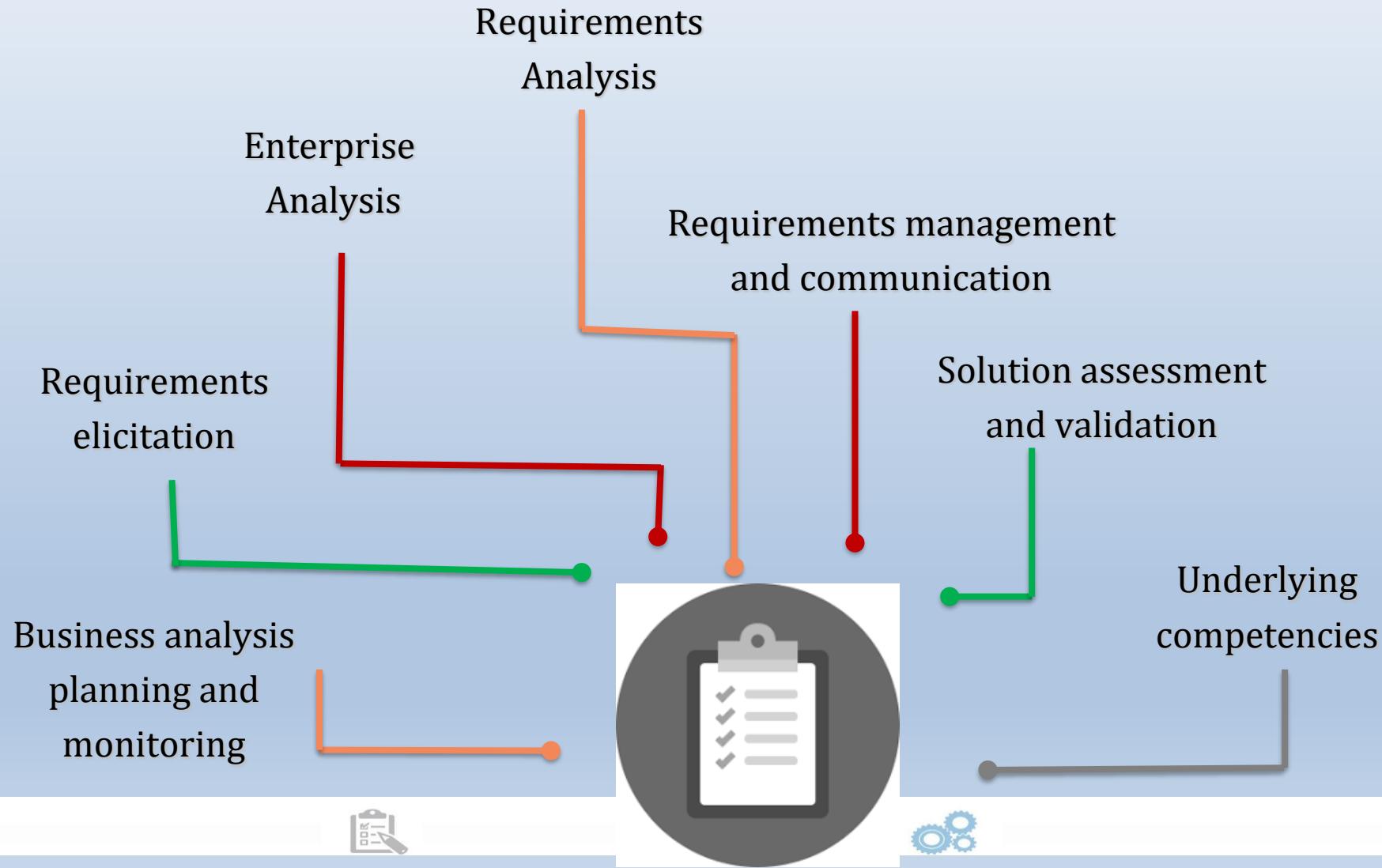


Knowledge areas define the set of techniques which a BA should understand and try to practice during the routine activities.



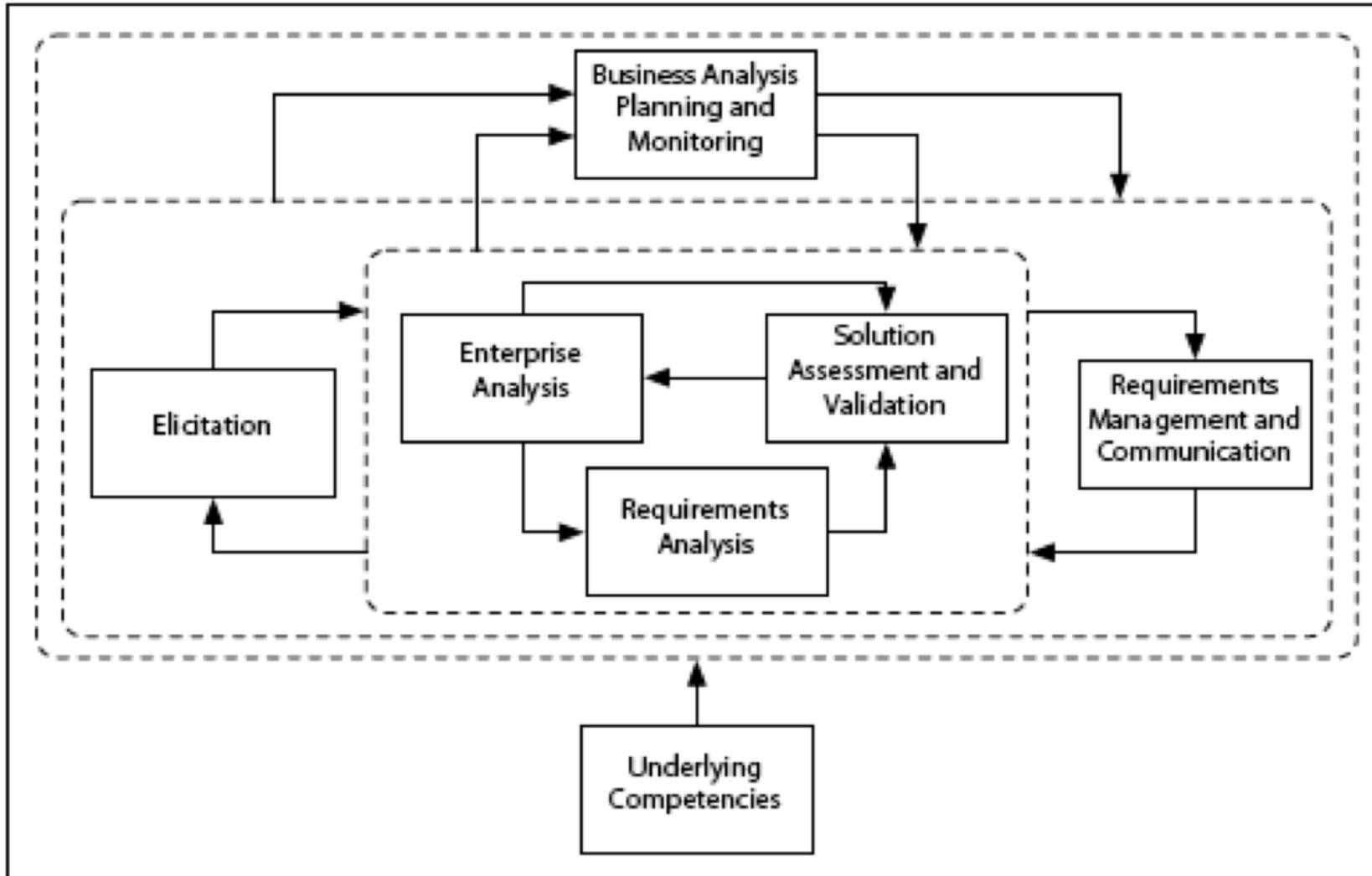
Knowledge Areas in the IIBA®'s BABOK® Guide

As per BABOK, the list of these knowledge areas (KA) is as below:

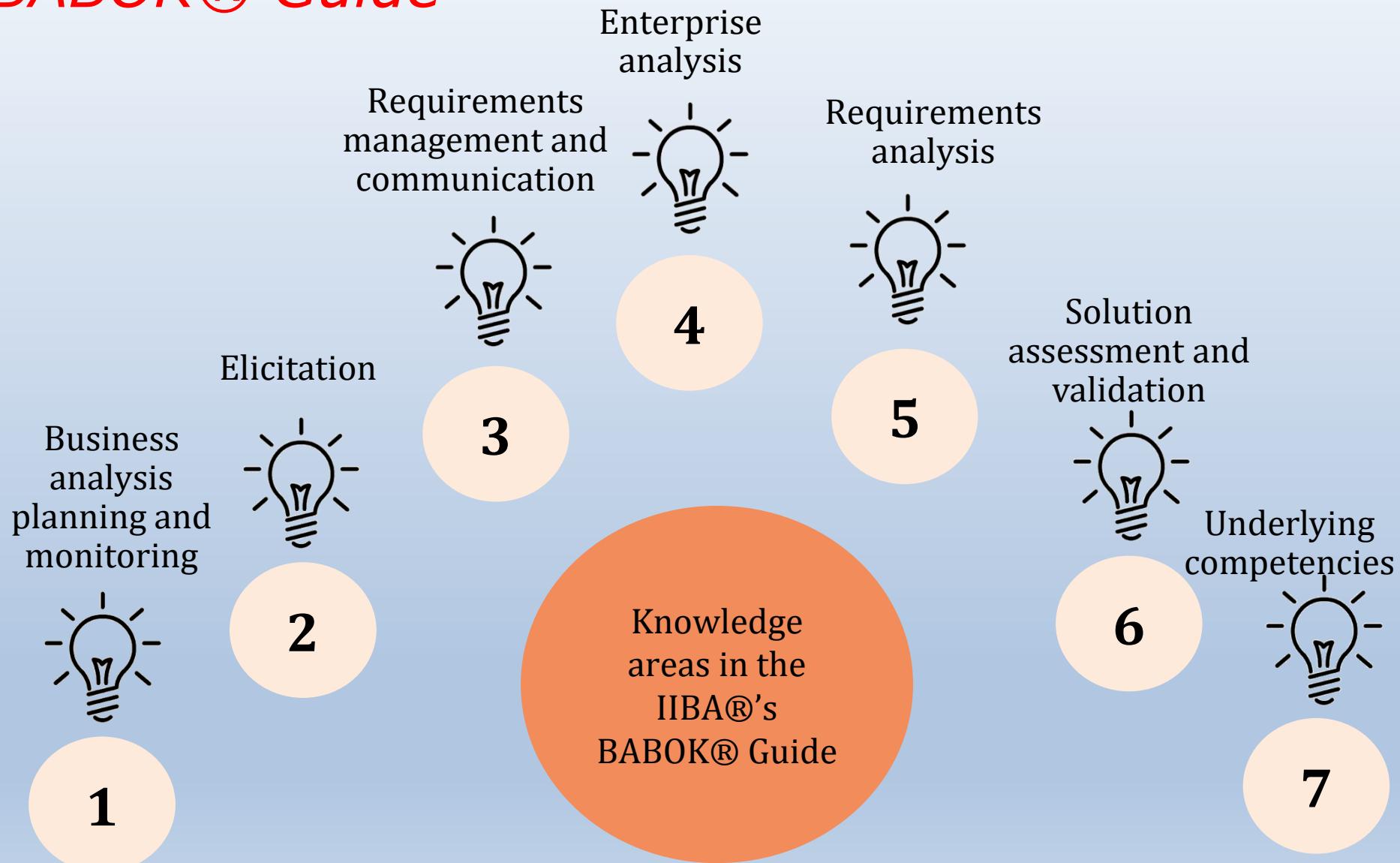


Knowledge Areas in the IIBA®'s BABOK® Guide

Figure 1–1: Relationships Between Knowledge Areas



Knowledge Areas in the IIBA®'s BABOK® Guide

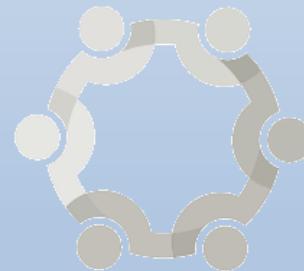


Business Analysis Planning and Monitoring

This knowledge area covers the activities which a business analyst should follow to determine the efforts involved in the future steps.

It mainly covers:

Stakeholder analysis



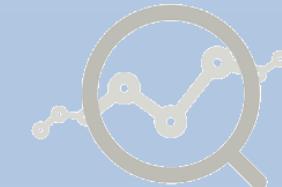
Managing risk,
issues and
requirements



Techniques to
manage the
requirements



Tracking of
the project
progress



This knowledge area is also called as the lead to all others as it governs the processes in the other knowledge areas.



S
u
j
a
t
a
B
a
t
r
a

Elicitation

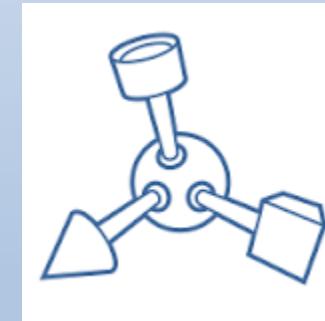
Requirements are the most important aspect of the project and understanding them correctly is of utmost importance.

It's not only important to gather the requirements from the available sources but also to elicit from them.

To collect the most precise information, there are several elicitation techniques followed by a BA like:



Brain storming



Prototyping



Interviewing

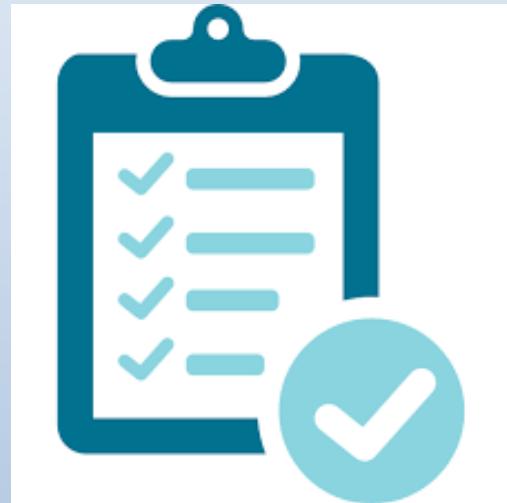


The actual purpose of the elicitation is to record the correct requirements of the stakeholders to avoid unwanted and superficial demands in future.



Requirements Management and Communication

There is no surety on the fact that the requirements which are elicited will not change and the business owners have got full rights in this world to change it at run-time.



Requirements



Conflicts

It is important for a business analyst to understand the severity of the requirements and manage them in such smart ways that the agreement on the elicited requirements remains on the same page and the conflicts, issues and changes can be minimized.

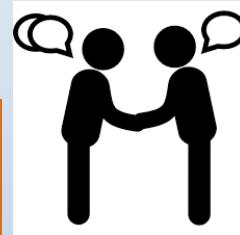


Requirements Management and Communication

This Knowledge area (KA) also covers the communication techniques of these requirements to all stakeholders.

This KA is all about:

Communicating requirements



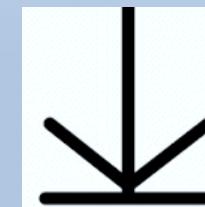
Resolving conflicts



Gaining formal approval



Base-lining



Tracking requirements



Enterprise Analysis

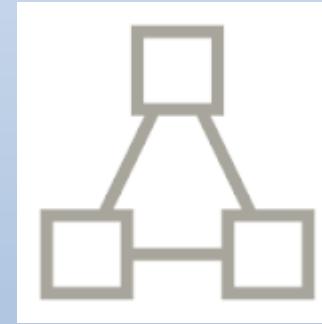
Enterprise Analysis is all about understanding the business need, comparing it with the existing functionality and finding the gaps in it.

Gaps

are also found in those areas which may stop the new business need which can stop its implementation.



Available Resources



Infrastructure



Techniques & Tools



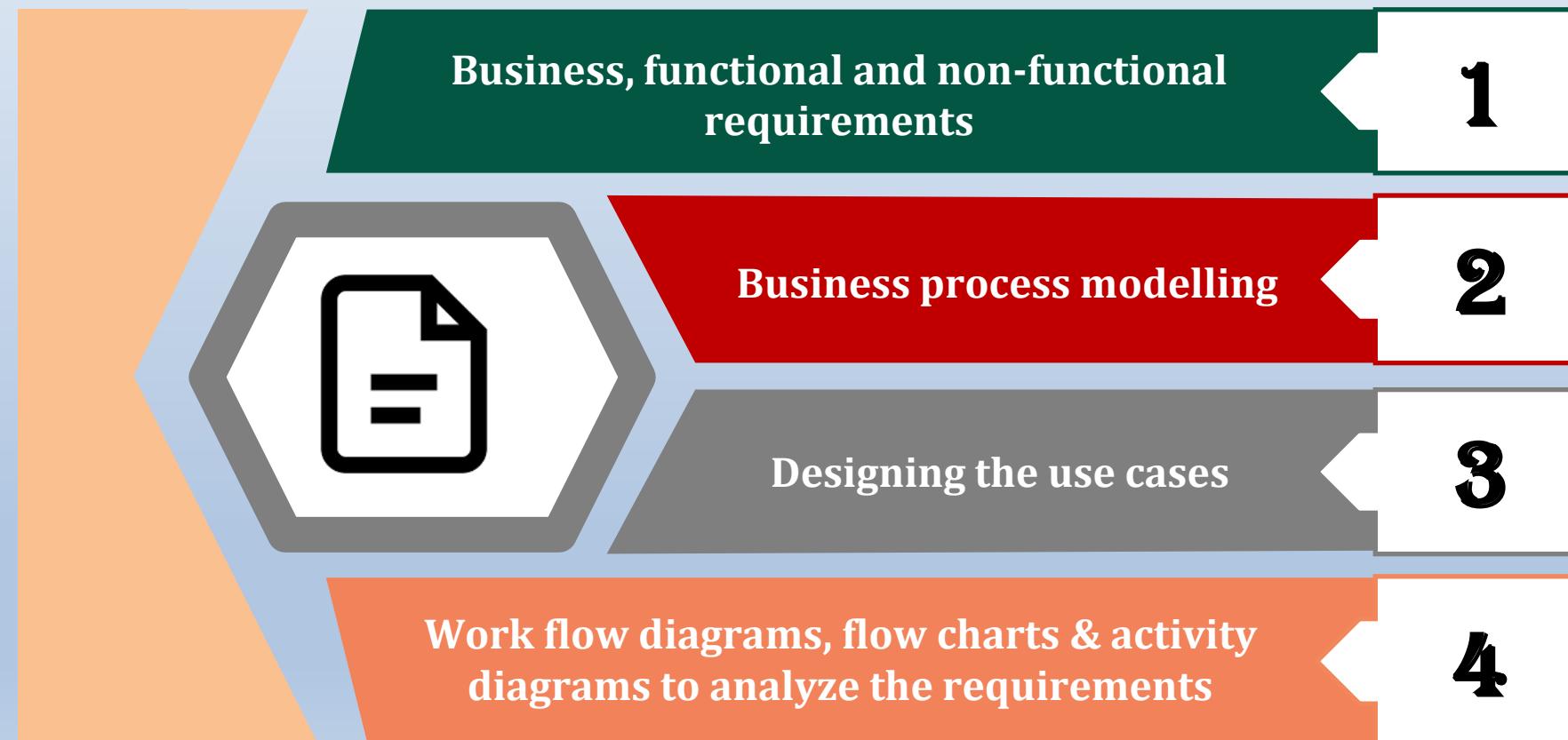
This KA thus, is all about managing those gaps, business case development and defining the solution scope.



Enterprise Analysis

The requirements gathered need to be prioritized and elaborated to the stakeholders in order to enable the development team to implement the same.

This KA covers the documentation of:



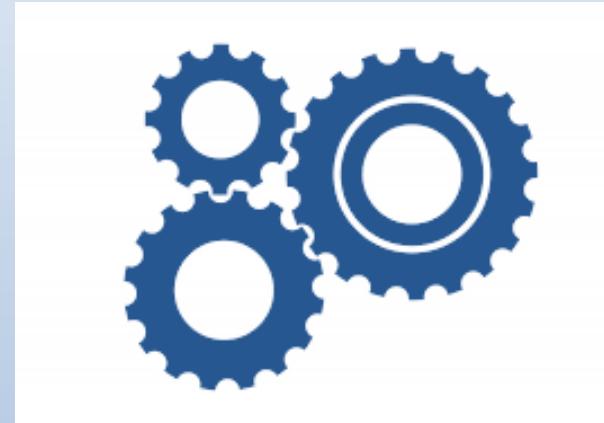
The validation and verification of the requirements are also covered under this KA.



Solution Assessment and Validation

Once the best solution is selected the BA makes sure that it meets those requirements throughout the lifecycle of the project.

This KA also covers the assessment of the:



02

Alternate available
solutions

02

Quality
assurance process

02

Pre and post
implementation processes



Underlying Competencies

Competencies which supports the BA to cover all the above mentioned KAs.

The personal skills of a BA that are covered under the umbrella of this knowledge area include:



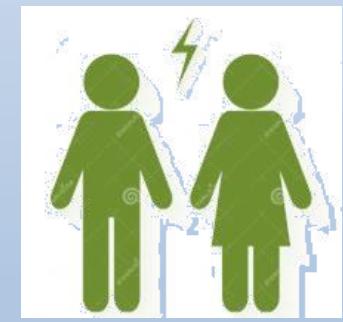
Leadership



Problem
Solving



Communication
Skills



Conflict
Management



BA Role in Project Life Cycle

Phase	Role
Initiation	<p>Defines a new project or a new phase of an existing project by obtaining authorization to start the project or phase. The business analyst may be involved in the following</p> <ul style="list-style-type: none">• Identify needs : Capture the business need and distinguish whether the need is a problem to solve or an opportunity to seize.• Defines goals & Objectives :Help the customer define goals and SMART objectives to allow the project success to be measured. Document in the form of business requirements.• Identify Stakeholders : Generally initiated by the sponsor and project manager (PM), however, the BA can refine the list or add stakeholders when appropriate.• Assess feasibility : Coordinate feasibility studies via a proof of concept to determine if the goals and objectives defined are possible prior to the allocation of resources.



S
u
j
a
t
a
B
a
t
r
a

BA Role in Project Life Cycle

Phase	Role
Planning	<p>Establishes the project, scope, refines the objectives, and defines the course of action required to reach those objectives. The business analyst would perform the following:</p> <ul style="list-style-type: none">• Determine Project Approach : Collaborate with the project team to determine the project approach based on the potential risks, timelines, constraints, information known, and stability of the business requirements• Business analysis plan : Create a business analysis approach based on the project approach. The plan will include an outline for various BA activities including stakeholder engagement and communication, requirements management, information management. This should be an input to the overall project plan.• Assess Current State : Determine the current situation via observation, document analysis or any other method that will provide context on existing processes and systems. The BA may document this in the form of a textual current state description or present as a process model.



BA Role in Project Life Cycle

Phase	Role
Planning	<ul style="list-style-type: none">• Root Cause Analysis - If the business need has been identified as a problem, perform root cause analysis in order to ensure the team is trying to solve the true source of the issue. This can be executed with a Fishbone Diagram or the Five Whys method.• Define Future State – Guide the customer to describe a definition of success, make decisions about the solution space, verify constraints, solidify business objectives, and determine the potential value. This may be documented in the form of a future state description or process model.• Assess Risk – Discuss the unknowns and work with the project manager and sponsor to determine the risk tolerance and risk management approach.• Perform gap analysis - Identify the difference between the current state and future state capabilities. This may be documented in the form of a process model illustrating what will change or the various transition states. This may also be represented as a change strategy.



S
u
j
a
t
a
B
a
t
r
a

BA Role in Project Life Cycle

Phase	Role
Planning	<ul style="list-style-type: none">Prepare for elicitation – Based on the change strategy, the BA will prepare to conduct various elicitation activities in an effort to get adequate details on the requirements.Provide estimates – Once the BA activities are planned out, the BA may provide estimates to the PM on the work effort as an input to the project plan.
Execute	The processes and activities performed to complete the work defined in the project management plan in order to satisfy the project requirements.

S
u
j
a
t
a
B
a
t
r
a



BA Role in Project Life Cycle

Phase	Role
Monitor And Control	<p>Processes required to track, review, and regulate the progress and performance of the project; identify areas in which changes to the plan are required, and initiate the related changes.</p> <ul style="list-style-type: none">•Control Scope – The BA plays a part in managing scope related to the <i>solution</i> by ensuring requirements are adequately traced and assessing requirements changes.
Close	<p>Performed to formally complete or close a project or contract.</p> <ul style="list-style-type: none">•Performance Measures – If customers don't have performance metrics related to the project readily available, the business analyst may be involved in collecting and evaluating performance measures to help determine the value and success of the project.•Lessons learned – Partake in lessons learned (retrospective) activities.



S
u
j
a
t
a
B
a
t
r
a

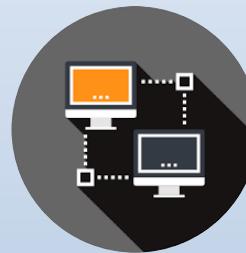
SDLC

The systems development life cycle (SDLC) is a term used in:

Systems
Engineering



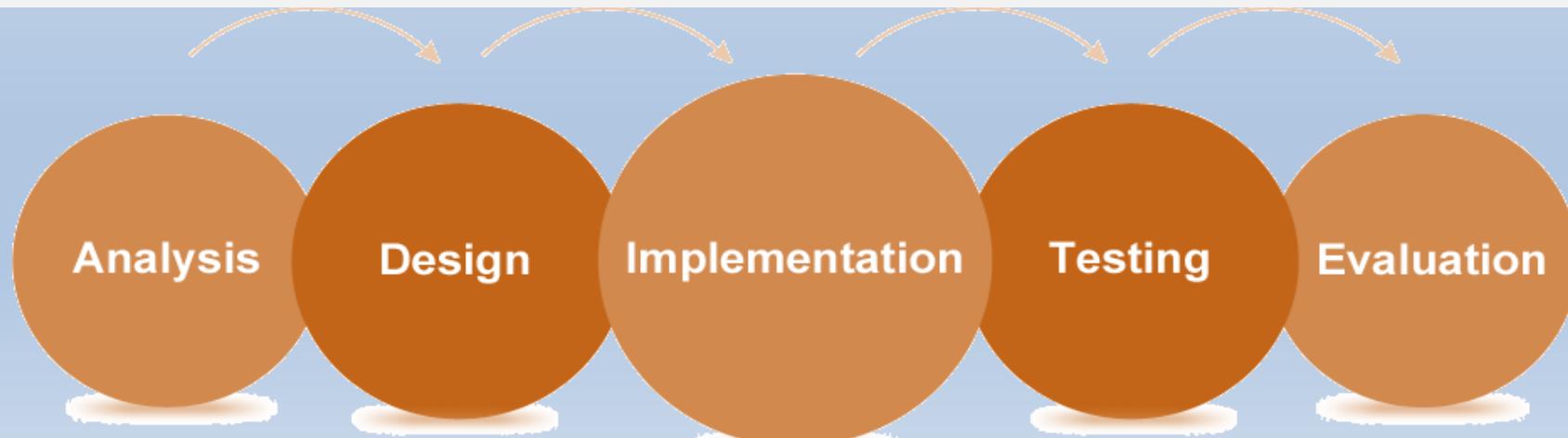
Information
Systems



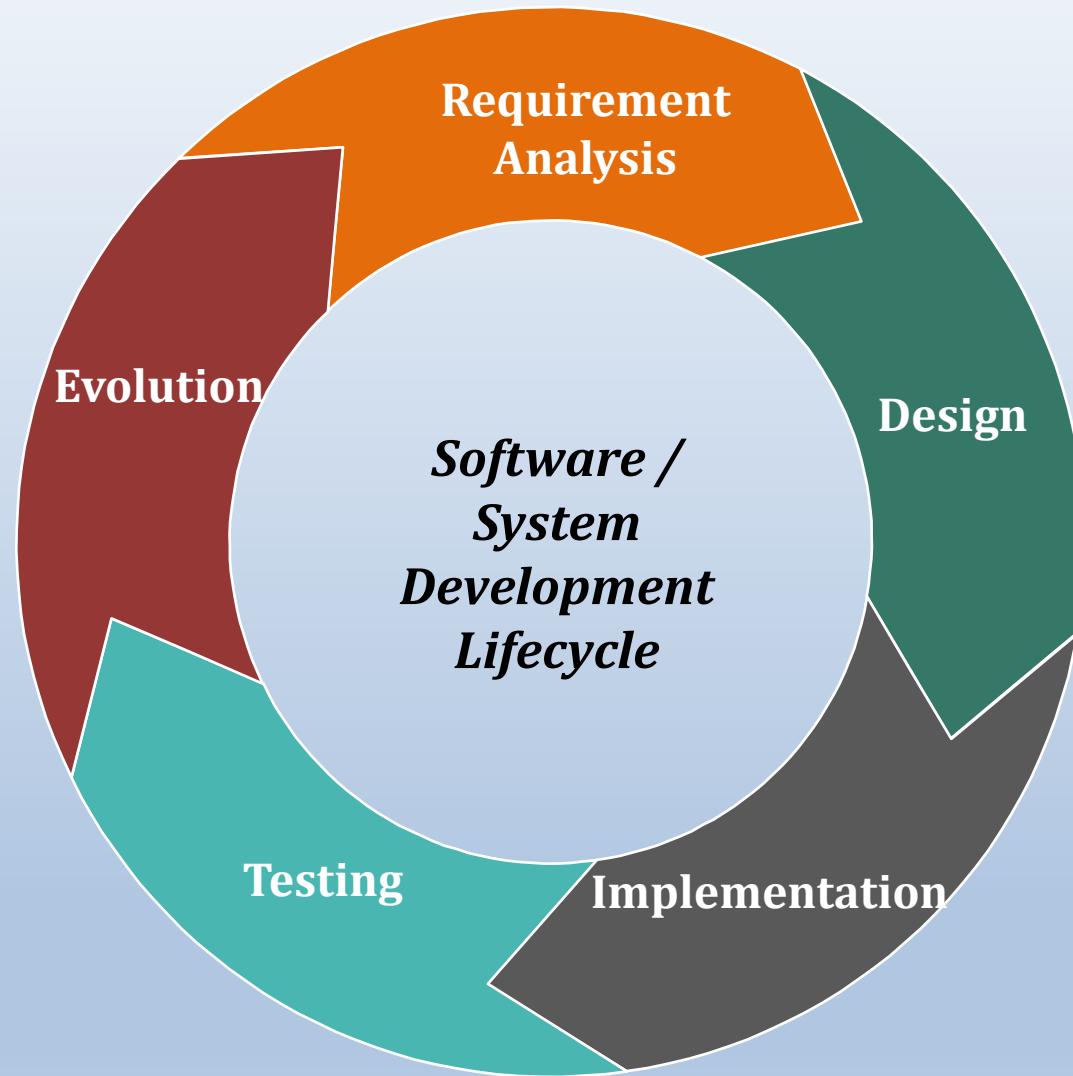
Software
Engineering



Also called application development life-cycle.

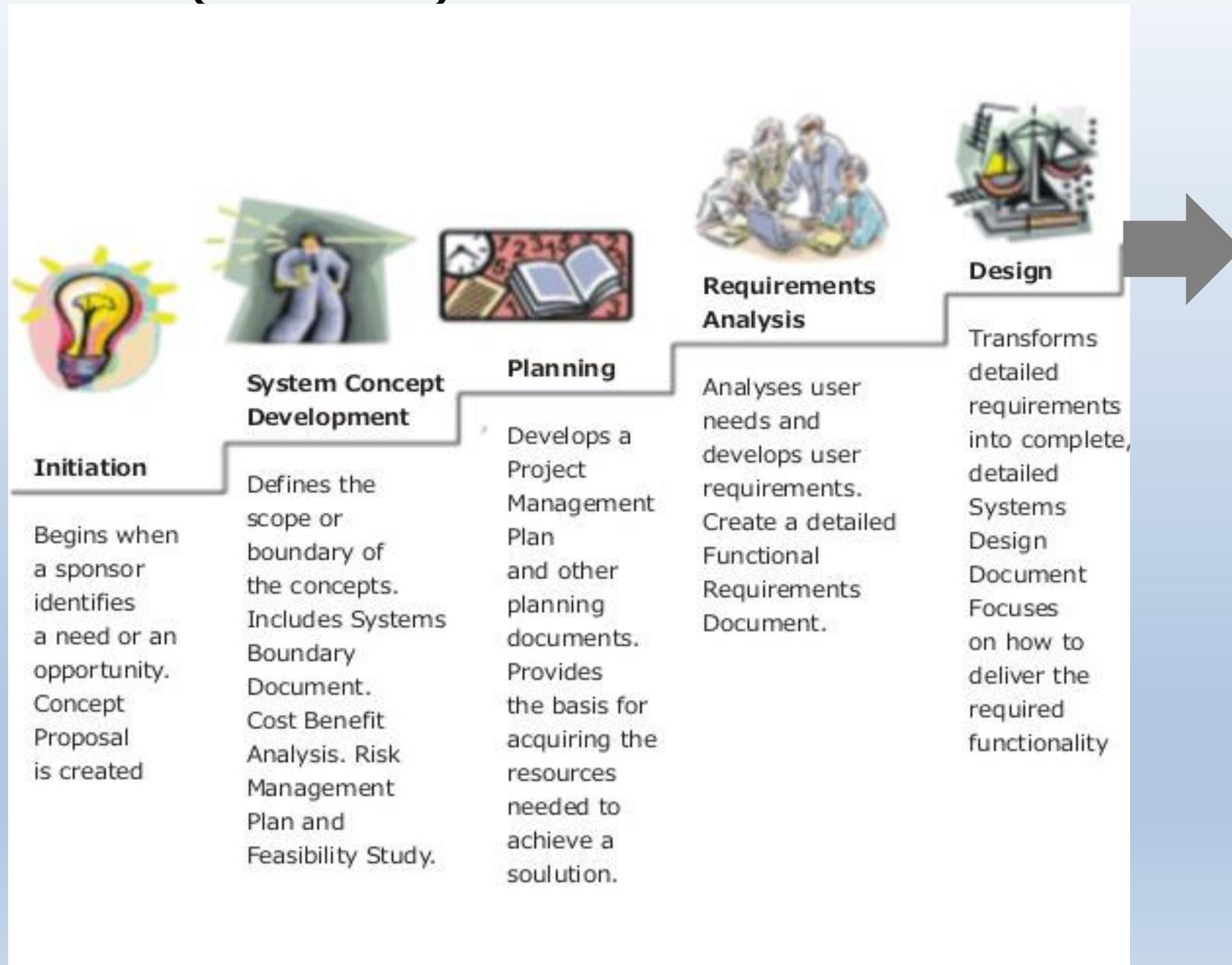


SDLC

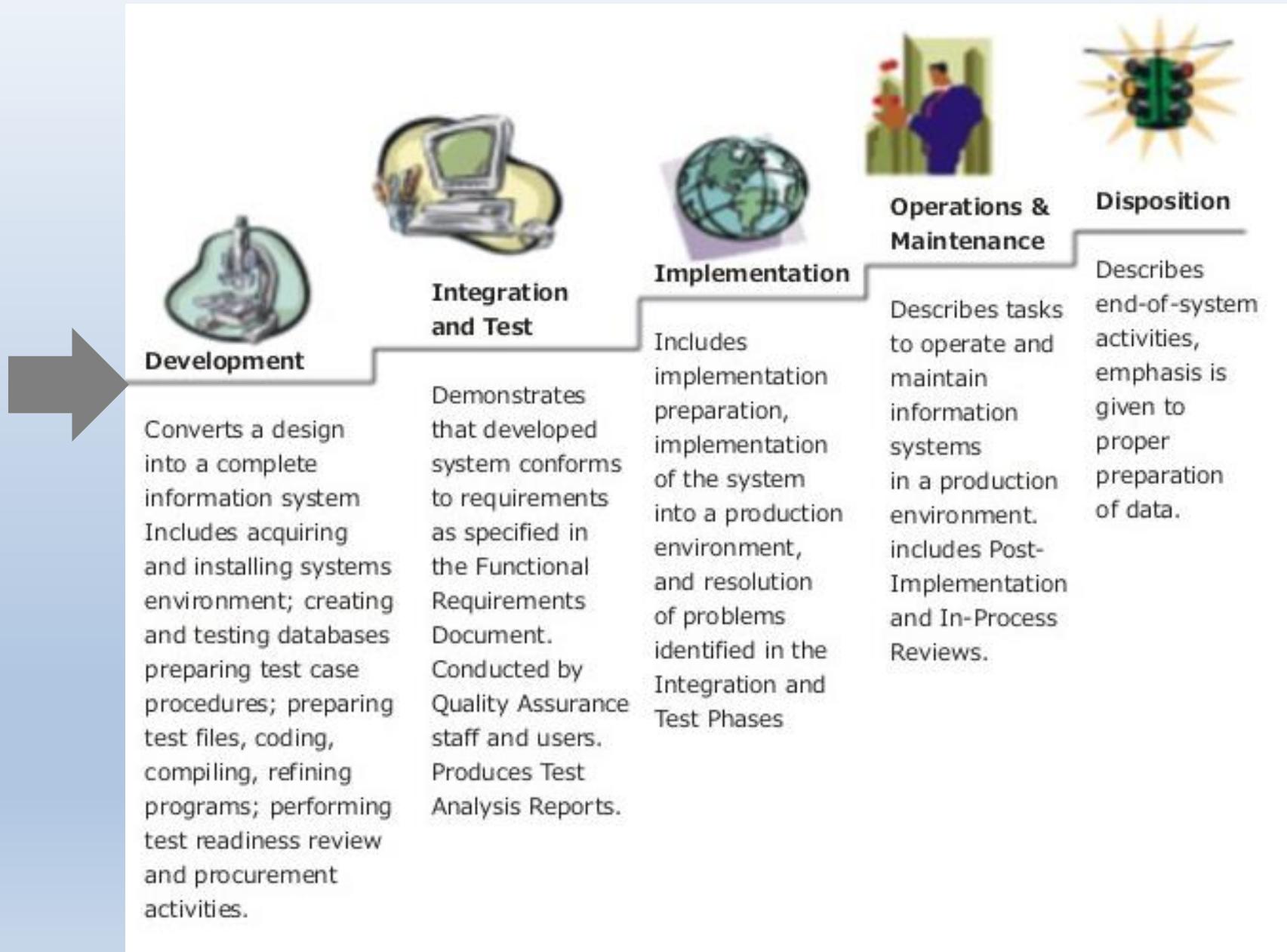


SDLC is the motherboard for all kinds of project developments!

SDLC Phases (Part 1 of 2)

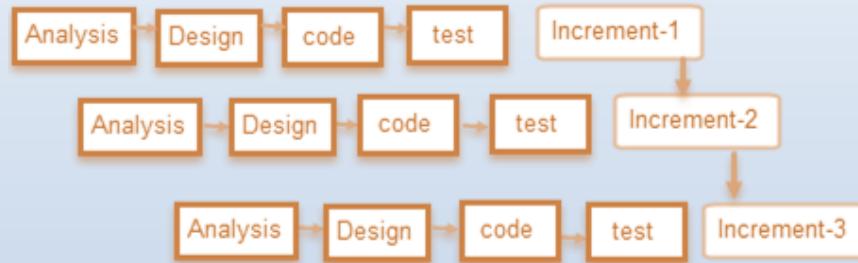


SDLC Phases (Part 2 of 2)

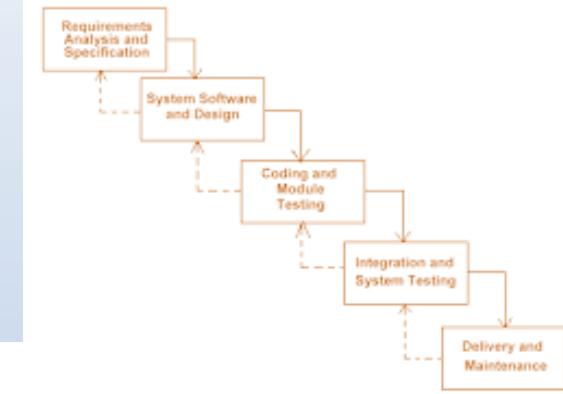


Types of SDLC

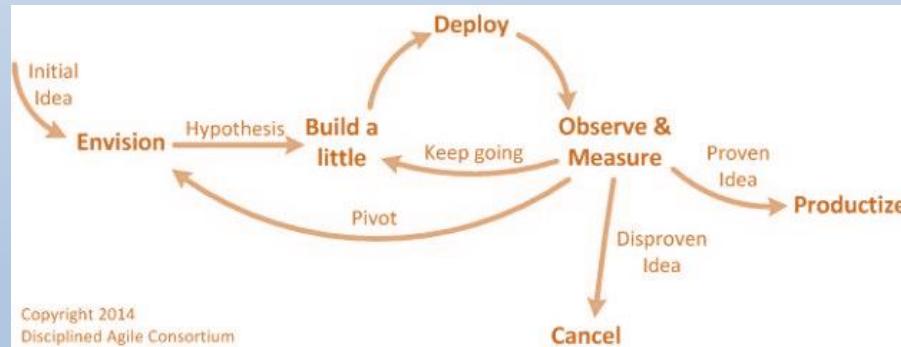
INCREMENTAL



WATERFALL



AGILE



ITERATIVE



Each kind of SDLC has its individual specificity tailored to situations where it may be necessitated.

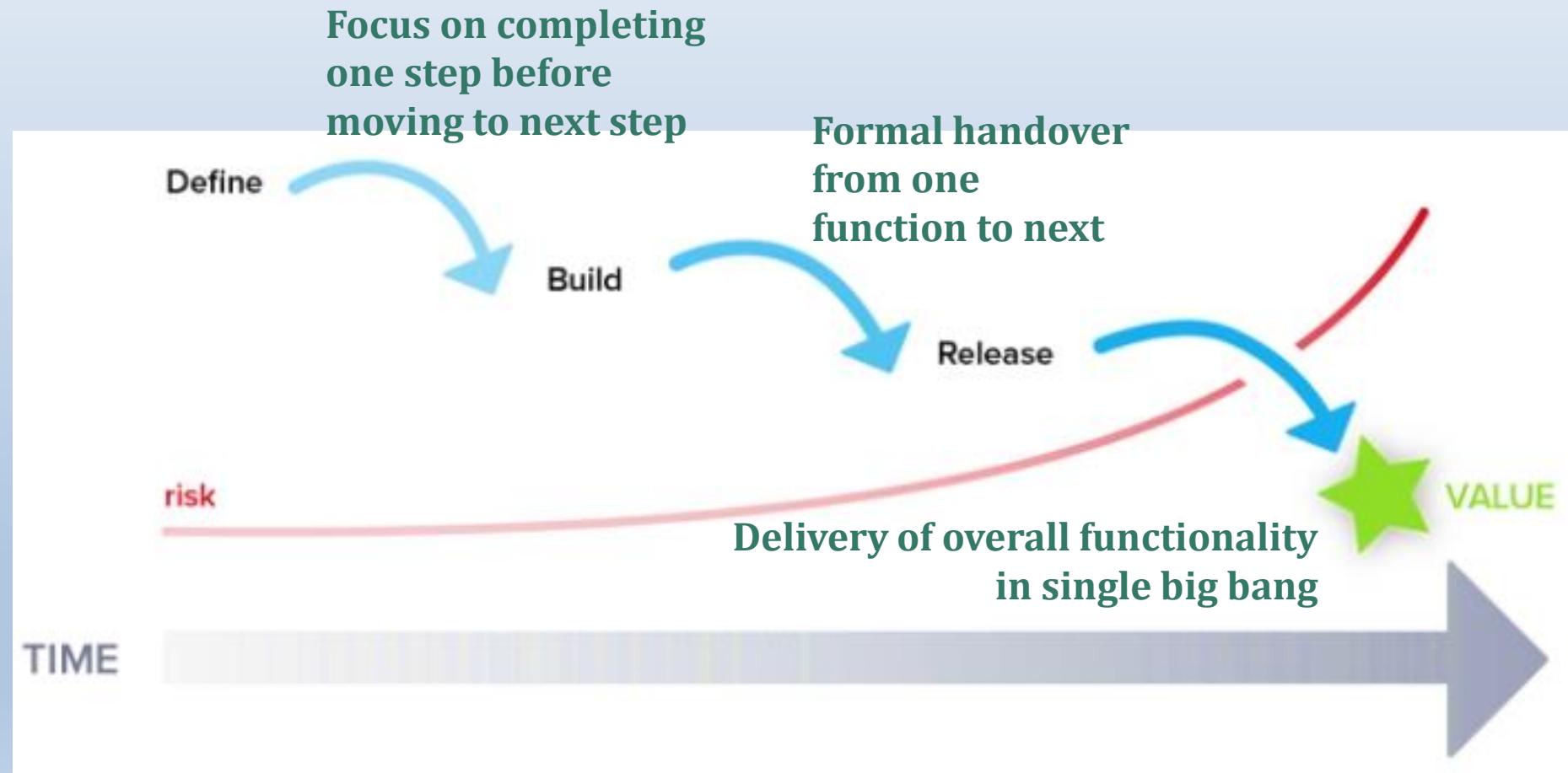
BA Role in SDLC Phases

Phase	Role
Planning or feasibility study	Studying existing system & analyzing
Analysis	<ul style="list-style-type: none">• Comprehend, and logically formalize business requirements.• Gather & analyze requirements.
Design	Designing the technical architecture & models
Development	Liaison with the developers
Testing	Test planning & testing
Implementation	Train the users on new system & monitor for initial teething problems.
Maintenance	Changing business requirements.

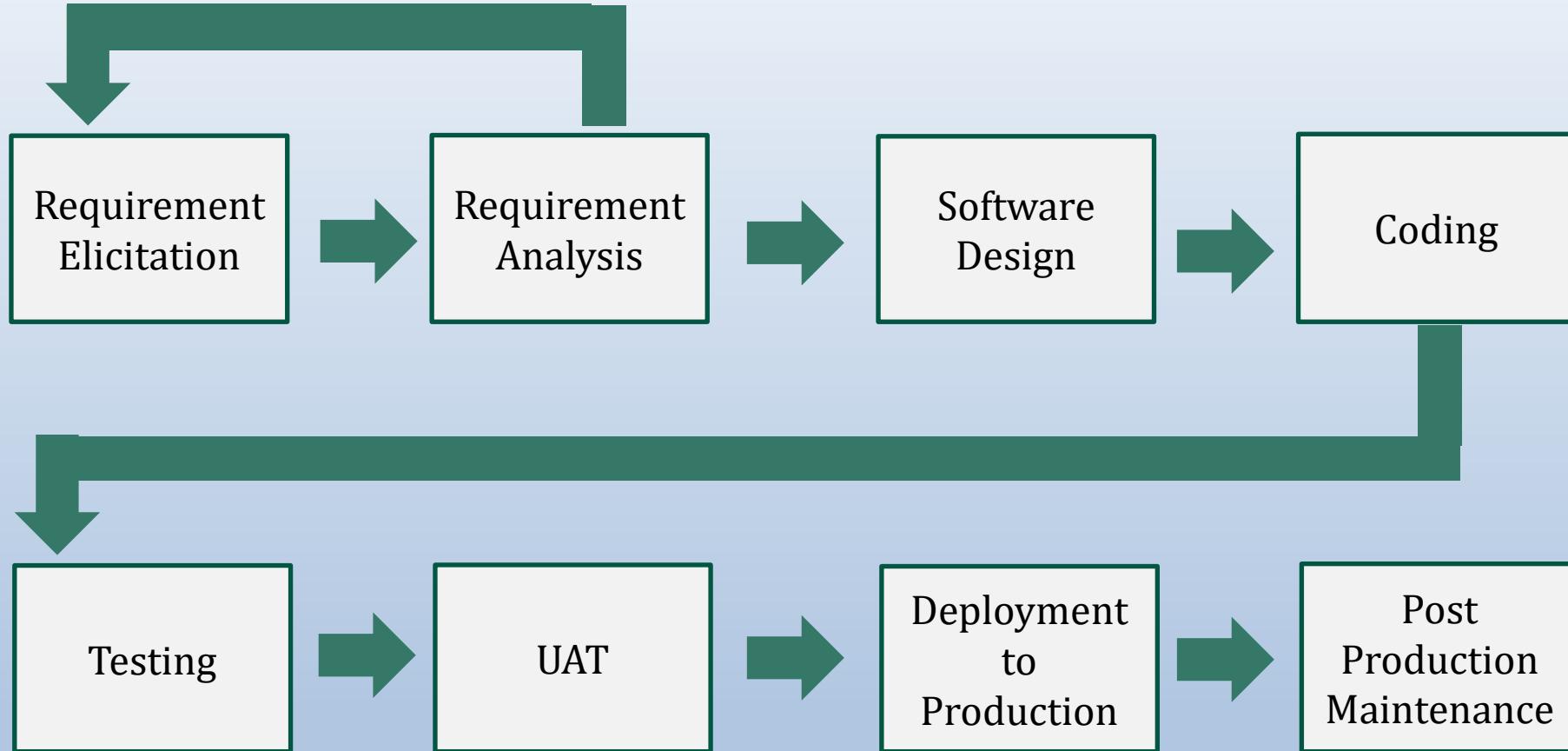
BA has practically a role in each phase of the solution, nurturing the same with diligence proves useful for the outcome!

Waterfall Model

Sequential phase driven approach.

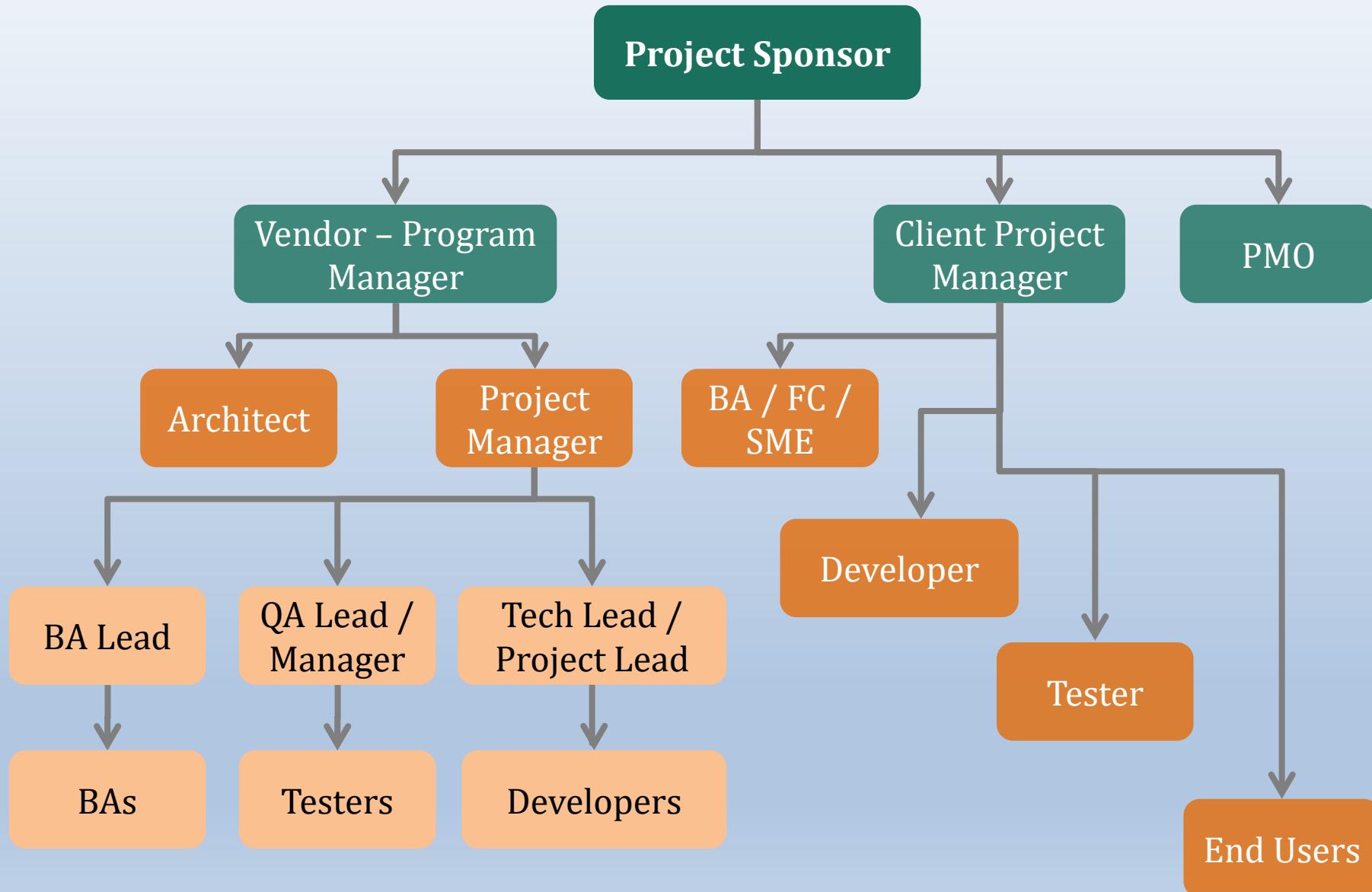


Waterfall Methodology (SDLC)



What should be the BA's responsibilities during each of the above phases?

Typical IT Waterfall Project Organization



Strengths of Waterfall Model

Easy to understand, easy to use

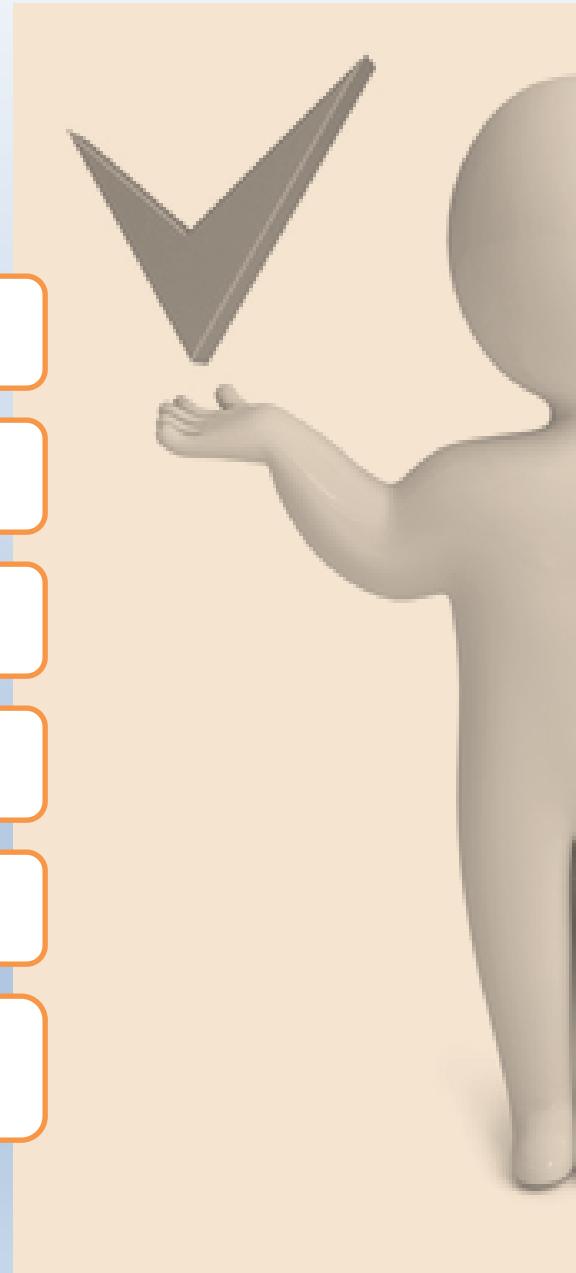
Provides structure to inexperienced staff

Milestones are well understood

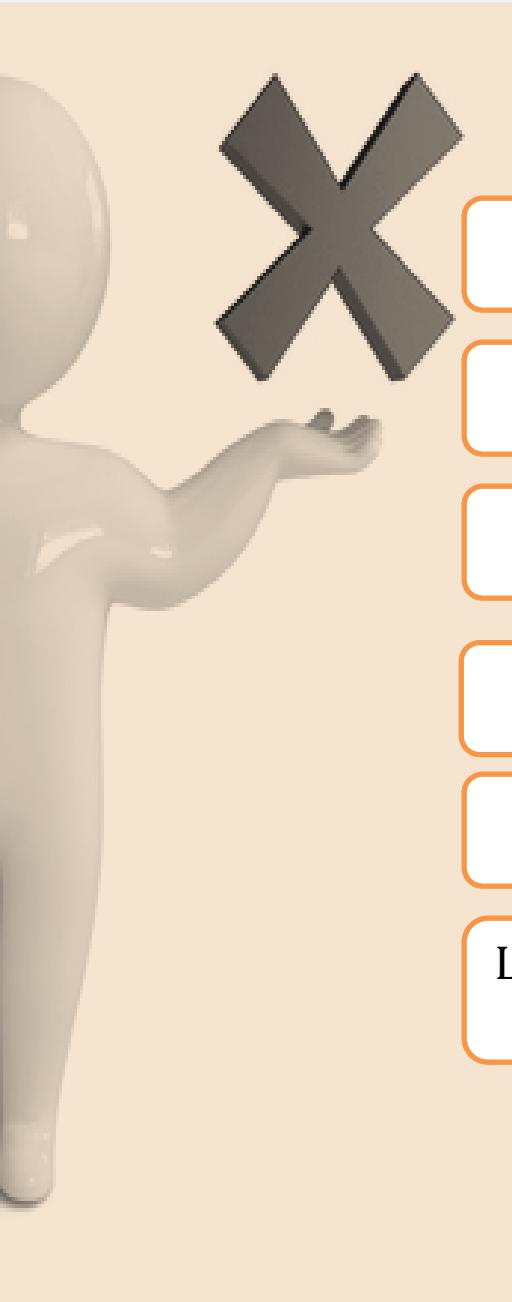
Sets requirements stability

Good for management control (plan, staff, track)

Works well when quality is more important
Than cost or schedule



Weaknesses of Waterfall Model



All requirements must be known upfront

Deliverables created for each phase are considered frozen – inhibits flexibility

Can give a false impression of progress

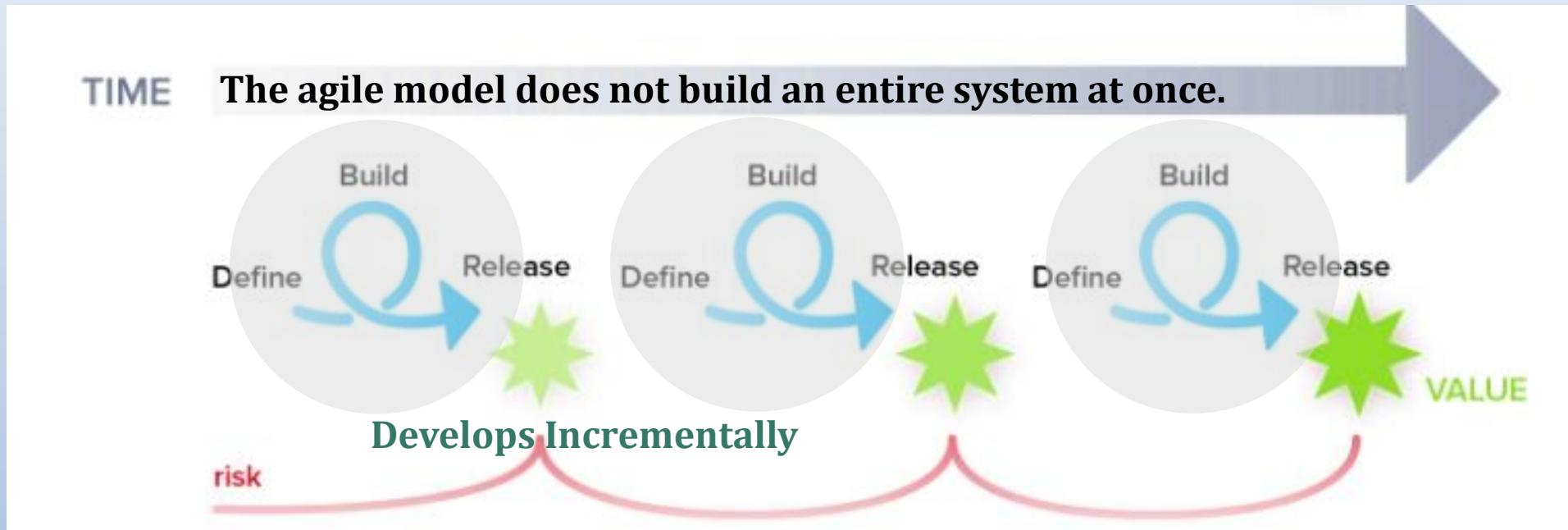
Does not reflect problem-solving nature of software development – iterations of phases

Integration is one big bang at the end

Little opportunity for customer to preview the system (until it may be too late)

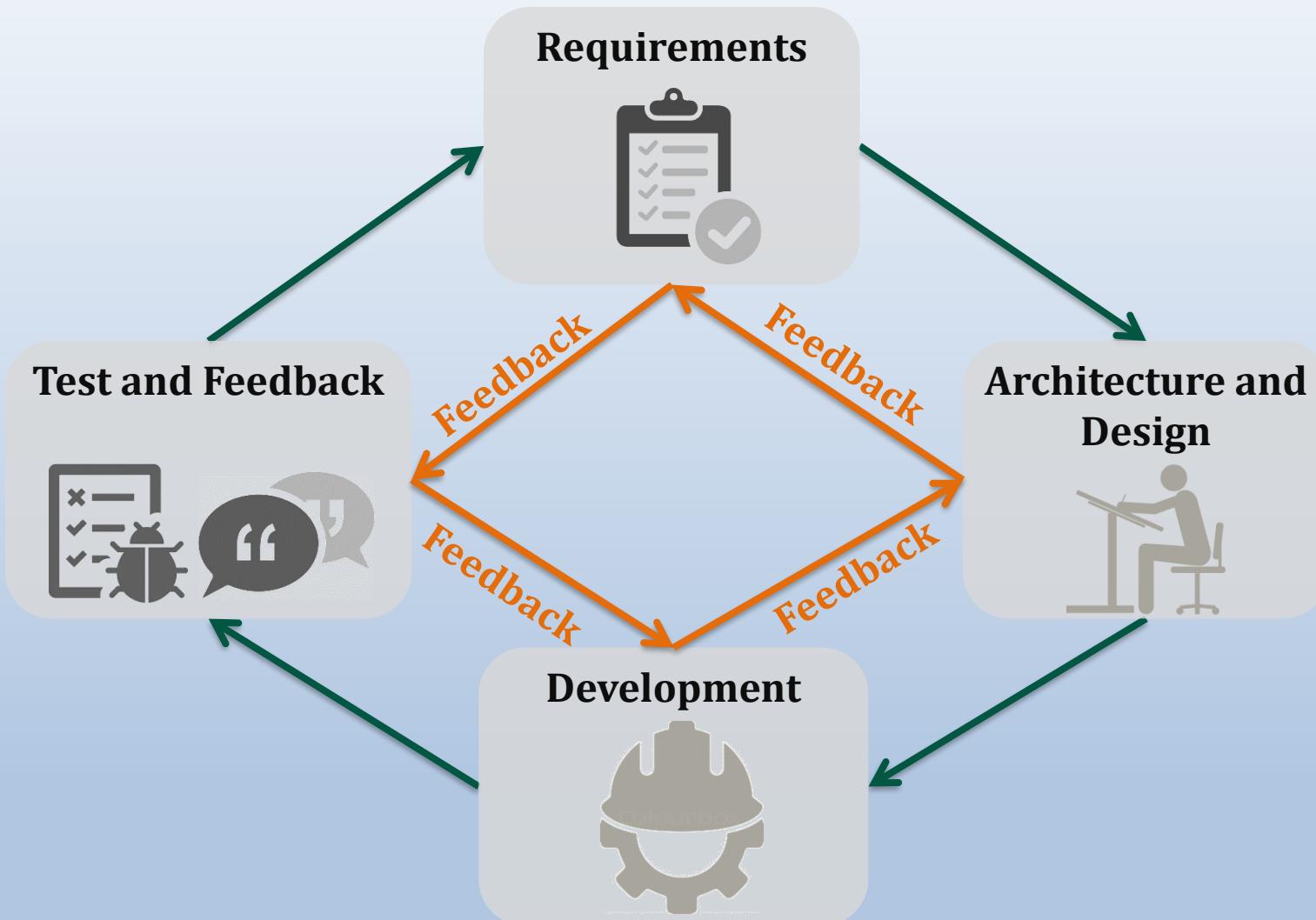
Adaptive or Agile Project Life Cycle

Less time is invested upfront for documenting requirements when development is done incrementally.



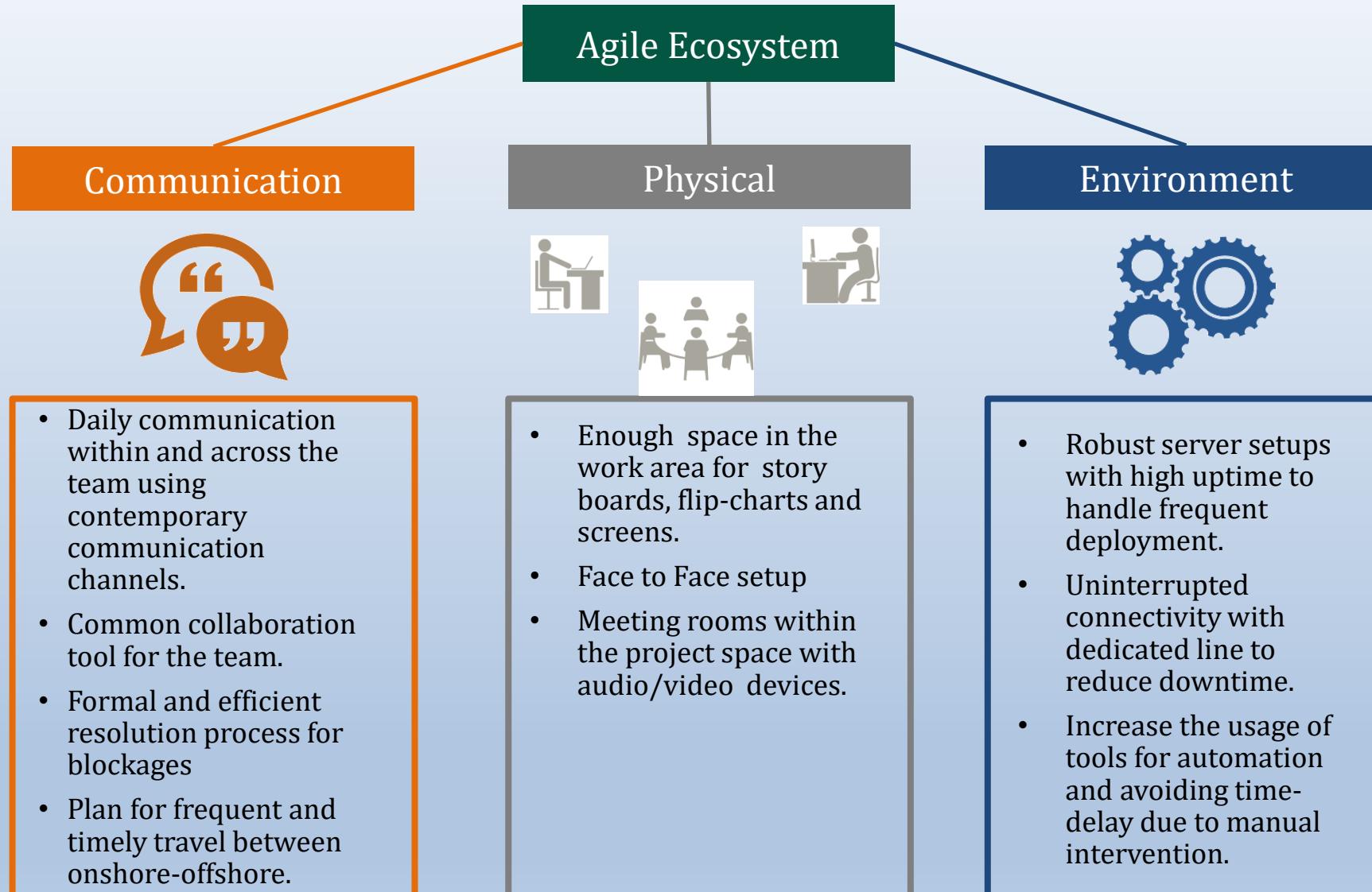
Unlike the more traditional waterfall approach, the agile development method is based on iterative and incremental development.

Adaptive or Agile Project Life Cycle



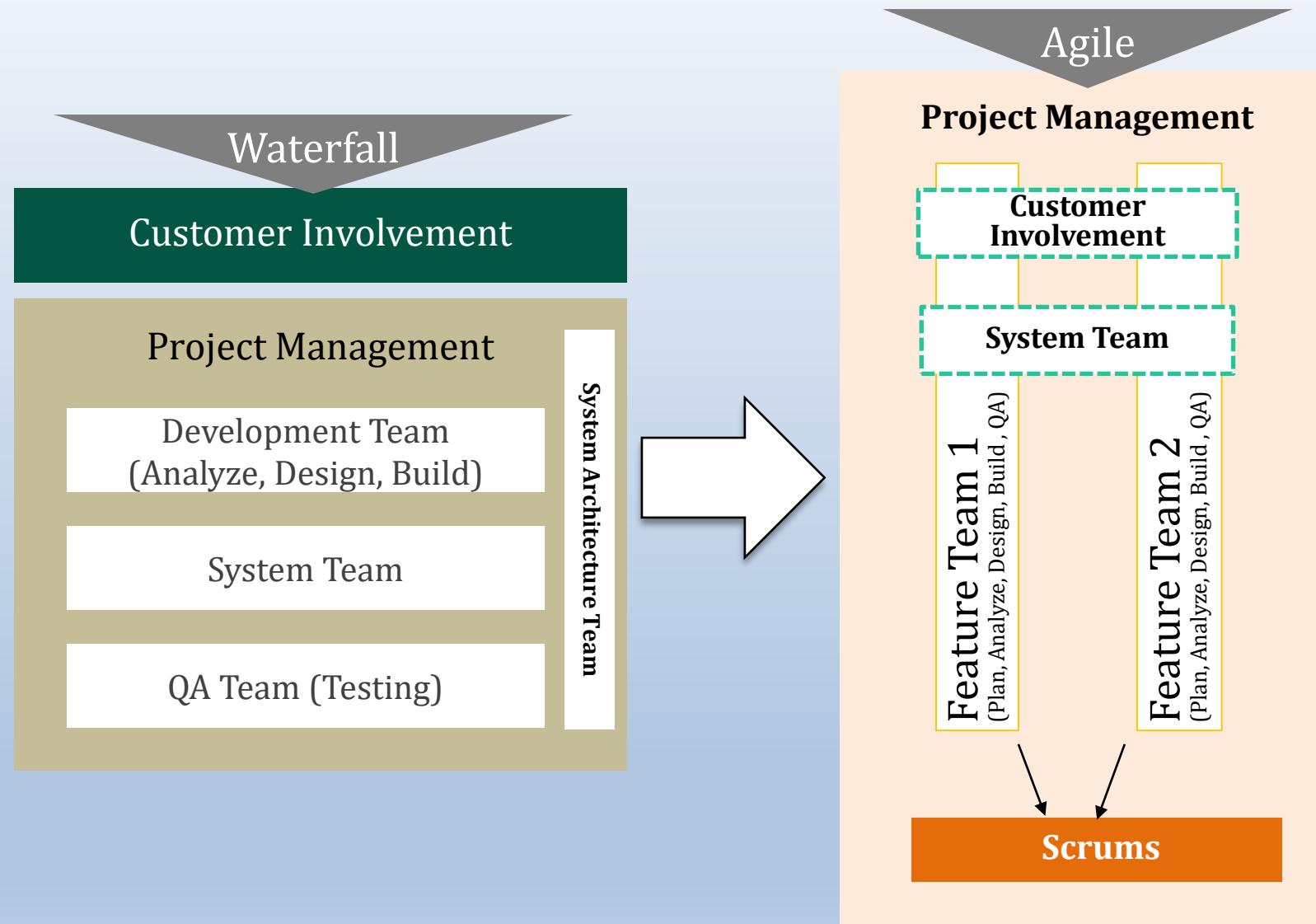
A mainline characteristic of agile software development is that customer feedback occurs simultaneously with development

Agile Ecosystem



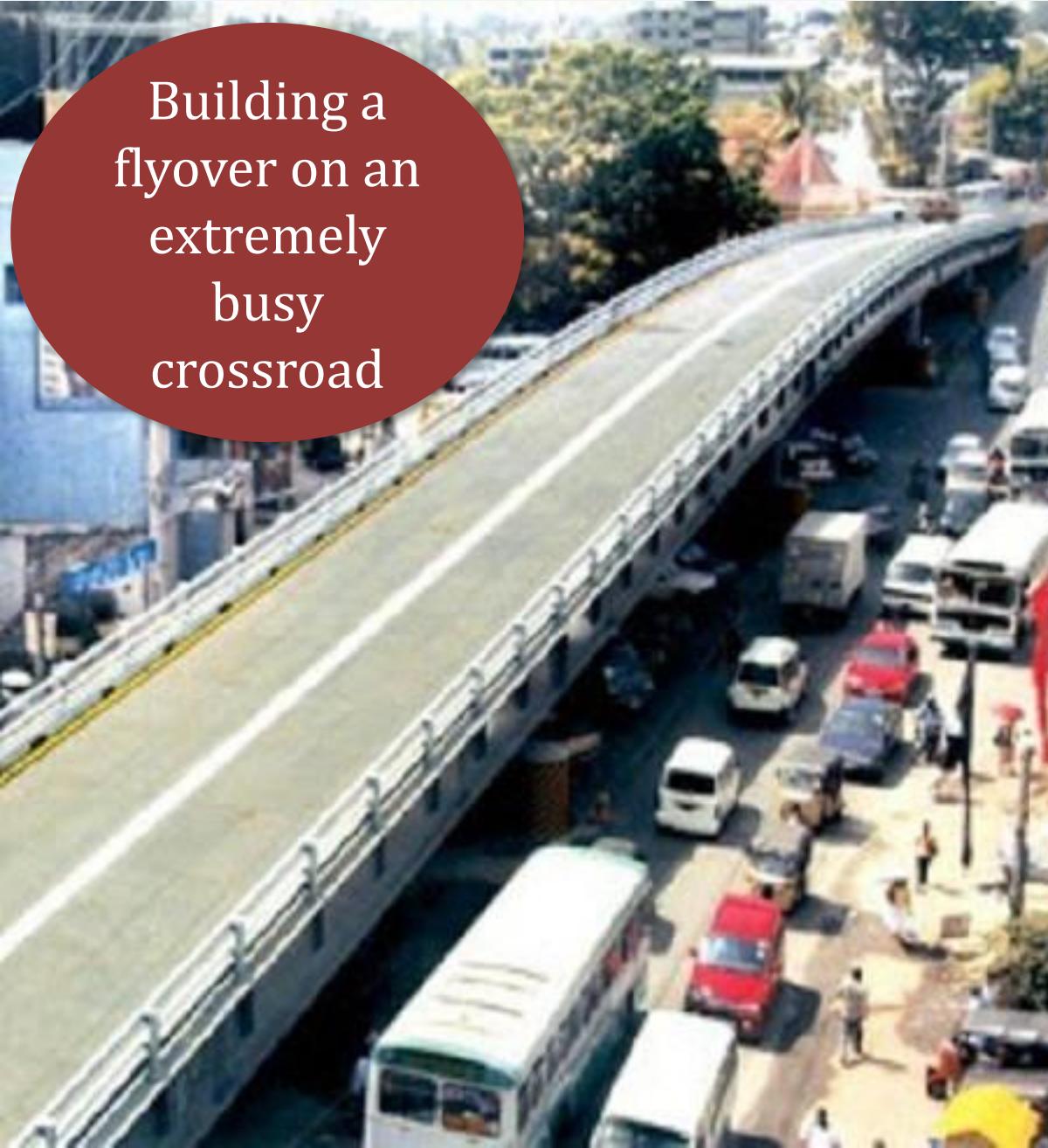
Agility is speed that's what matters when you need
to reduce the speed to market!

Team Transformation



Waterfall models get transmuted into Agile model as more and more traditional set ups are incorporating some or most of agile within themselves

Example of Agile Approach



- This flyover project demonstrated how incremental delivery can indeed be extremely useful for the project as well as for the end customers.
- The construction was planned to have incremental delivery, so that one direction of the flyover would be constructed before starting the work on the second direction

Example of Agile Approach



The one-way flyover construction is completed and opens for two-way traffic

01

The overall traffic is still slow, but much better than without any flyovers.

02

Here the end customer (commuter) is using what we call a product of incremental delivery.

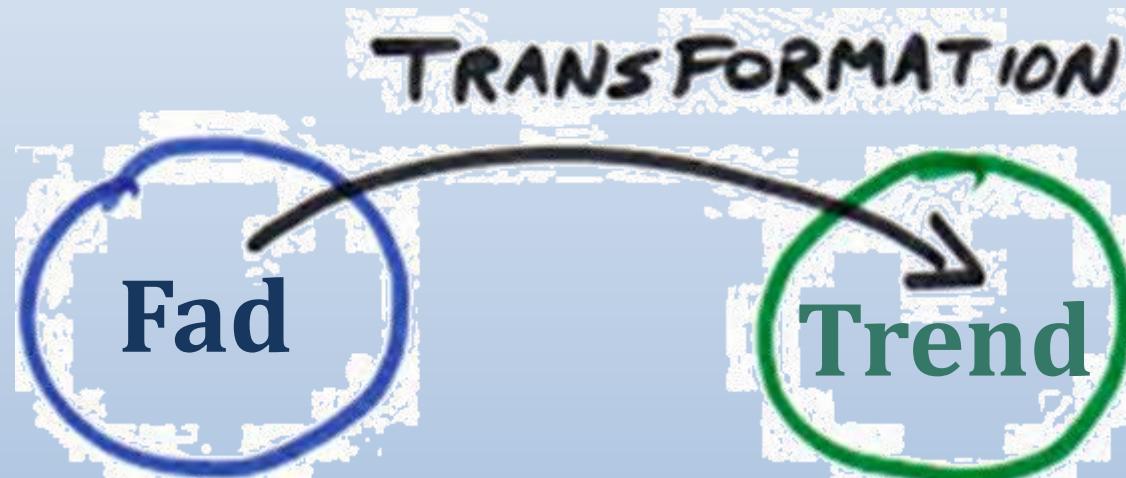
03

This incremental delivery helped customers use the project (the flyover) in nine months instead of waiting twice that long (plus some inevitable delays).

Agile Software Development

Agile is one of the big buzzwords of the IT development industry.

Five years ago,
agile practices transformed from the latest fad to a respectable trend.



As of **2016**, the majority of business analysts we have are experienced or are working in agile teams.

That's because agile is much more widely accepted and adopted now as a discipline.

Agile Software Development

Agile development is a different way of managing IT development teams and projects.

The traditional approach to managing software development projects was failing far too often and there had to be a better way.

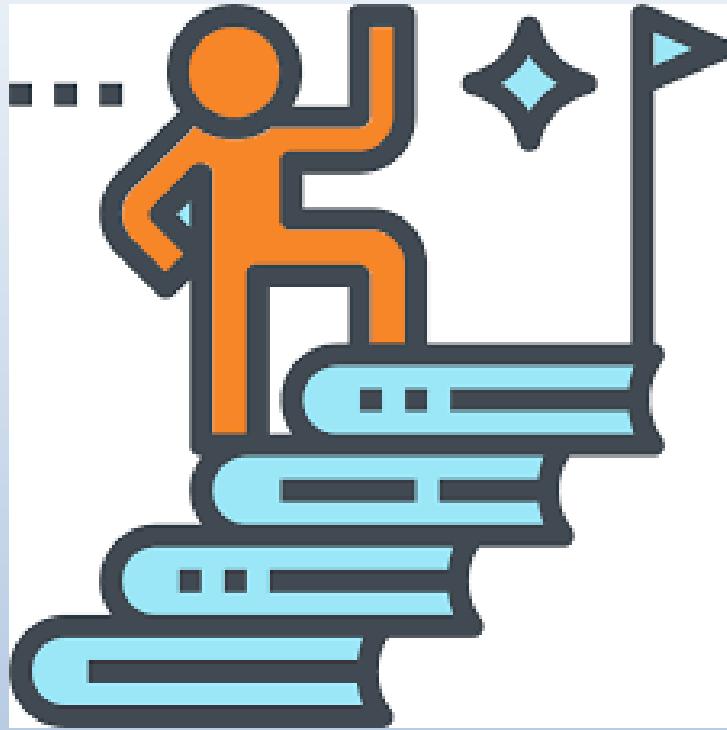
The agile manifesto describes 4 important values that are as relevant today as they were then.

It says,

“We value individuals and interactions over processes and tools
Working software over comprehensive documentation
customer collaboration over contract negotiation
responding to change over following a plan”.

Agile Software Development

Over the last
10 years

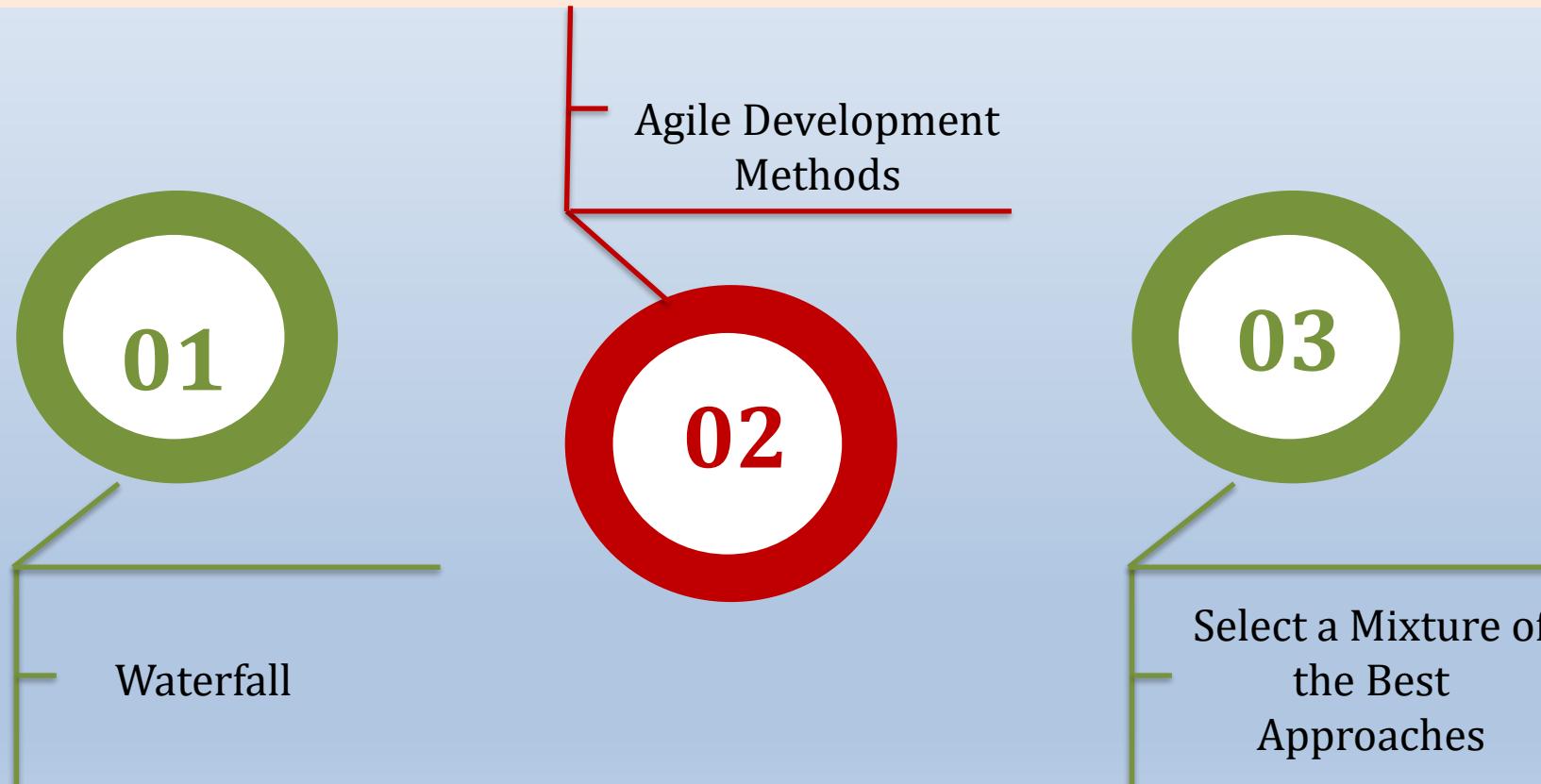


There is an ever-increasing volume of success stories, where companies have dramatically improved the success and performance of their IT development teams and projects.

Agile Software Development

Agile is not a magic bullet for all software development issues.

The real trick is to know lots of techniques from various :



To do this reliably with any degree of success really requires a lot of experience and skill.

Champion of Change - The Business Analyst

1

Agile methods break the product into small incremental builds.

2

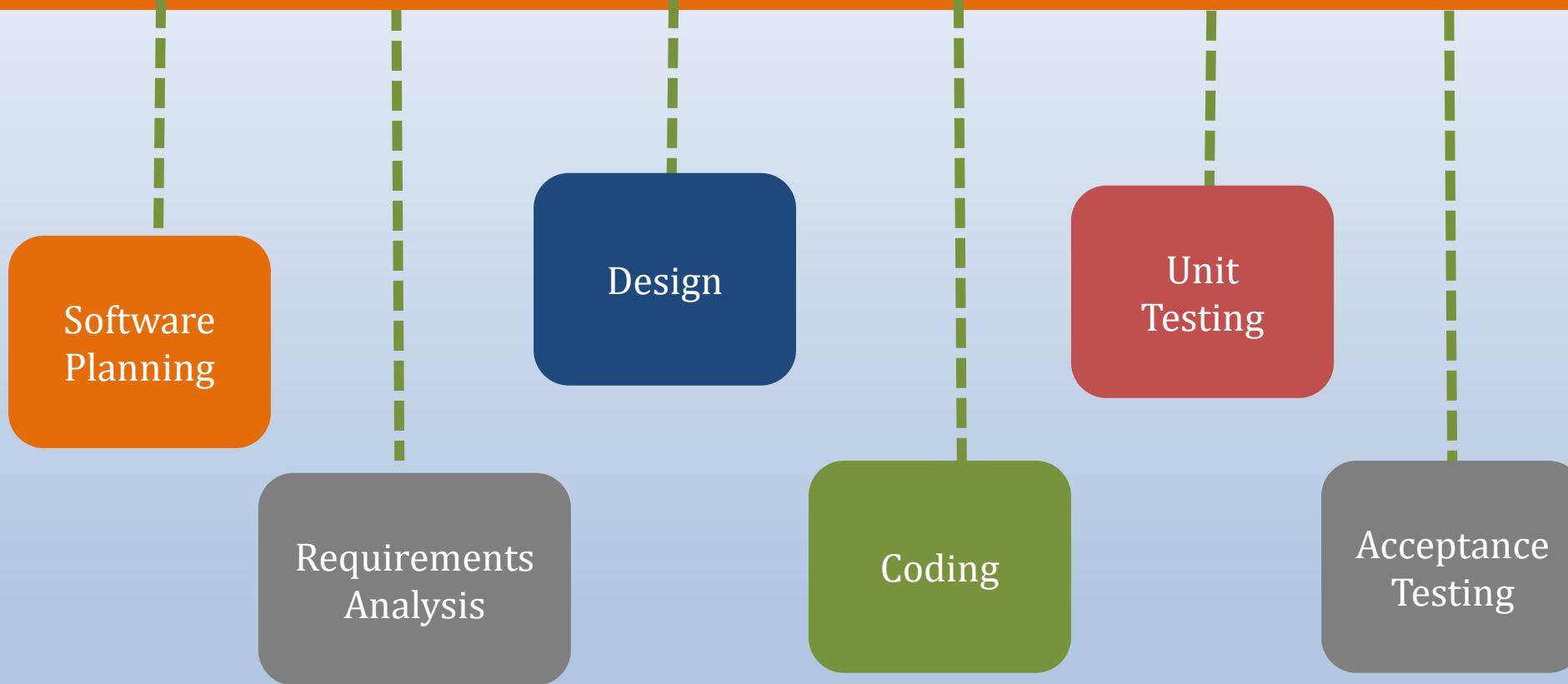
These builds are provided in iterations.

3

Each iteration typically lasts from about one to three weeks.

Agile Software Development

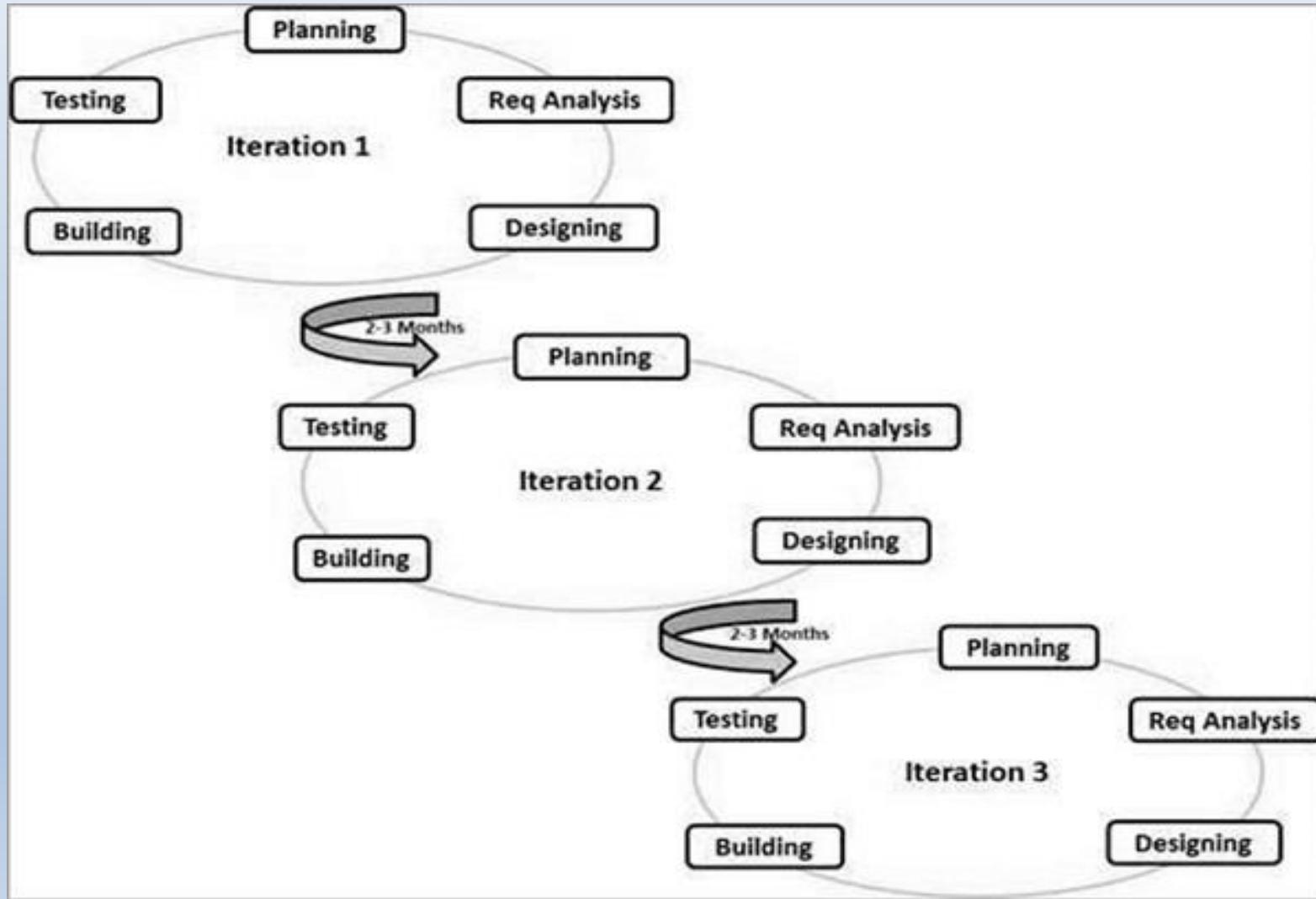
Every iteration involves cross functional teams working simultaneously on various areas like -



At the end of the iteration, a working product is displayed to the customer and important stakeholders.

Agile Software Development

Here is a graphical illustration of the Agile Model –



Agile Software Development

The Agile thought process had started early in the software development and started becoming popular with time due to its flexibility and adaptability.

Agile methods :



1

Break tasks
into small
increments



2

Involve
minimal
planning



3

On issue's vital to
organisational's
welfare

Do not Directly Involve Long-term Planning

Iterations are short time frames that last from one to four weeks.

Agile Software Development

Each iteration involves a team working through a full software development cycle, including:



Planning



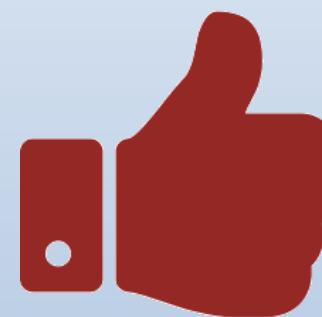
Requirements
Analysis



Coding



Unit
Testing



Acceptance

Agile Software Development

This minimizes overall risk and allows the project to adapt to changes quickly.

An iteration might not add enough functionality to warrant a market release, but the goal is to have an available release (with minimal bugs) at the end of each iteration.



Multiple iterations might be required to release a product or new features.

Agile methods emphasize face-to-face communication over written documents when the team is all in the same location.

Features of Agile

Principle 1: Active user involvement is imperative

Active user involvement is the first principle of agile development.

External users cannot be involved in project development projects



External Customers



Project Development

In this event it is imperative to have a senior and experienced user representative involved throughout.

Features of Agile

Principle 2: Agile Development Teams Must Be Empowered

An agile development team must include all the necessary team members to make decisions and make them on a timely basis.



The team must establish and clarify and prioritise requirements, agree to the tasks required to deliver, and estimate the effort involved.

Features of Agile

Principle 3:
Requirements evolve, but timescales are fixed.

01

Agile development works on the premise that requirements emerge and evolve, and that however much you analyze and design, this will always be the case because you cannot really know for sure what you want until you see and use the software.

02

In the time, you would have spent analyzing and reviewing requirements and designing a solution, external conditions could also have changed.

03

Agile development projects accept change; in fact, they expect it.

04

In these projects, requirements are allowed to evolve, but the timescale is fixed.

Features of Agile

Principle 3:
Requirements evolve, but timescales are fixed.

05

To include a new requirement, or to change a requirement, the user or product owner must remove a comparable amount of work from the project in order to accommodate the change.

06

This ensures the team remains focused on the agreed timescale and allows the product to evolve into the right solution.

07

It does, however, also pre-suppose that there's enough non-mandatory features included in the original timeframes to allow these trade-off decisions to occur without fundamentally compromising the end product.

Features of Agile

Principle 4: Agile Requirements are barely sufficient

Agile development teams capture requirements at a high level and on a piecemeal basis, just-in-time for each feature to be developed.



Agile requirements are ideally visual and should be barely sufficient, i.e. the absolute minimum required to enable development and testing to proceed with reasonable efficiency.



The rationale for this is to minimise the time spent on anything that doesn't actually form part of the end product.



Features of Agile

Principle 5: Done means done!



Features developed within iteration i.e. a sprint in scrum, should be 100% complete by the end of the sprint.



Too often in software development, “done” doesn’t really mean “done!”, tested, styled and accepted by the product owner. It just means developed.



Make sure that each feature is fully developed, tested, styled, and accepted by the product owner before counting it as “DONE!”.



If there is any doubt about what activities should or shouldn’t be completed within the sprint for each feature, “DONE!” should mean shippable.

Features of Agile



Multiple features can be developed in parallel in a team situation.



However, within the work of each developer, do not move on to a new feature until the last one is shippable.



This is important to ensure the overall product is in a shippable state at the end of the sprint, not in a state where multiple features are 90% complete or untested, as is more usual in traditional development projects.

Features of Agile

Principle 6: Agile testing is not for dummies!

Testing is integrated throughout the software development lifecycle.

Agile development does not have a separate test phase as such.



Developers are much more heavily engaged in testing, writing automated repeatable unit tests to validate their code.

Features of Agile

Principle 6: Agile testing is not for dummies!



1

With automated repeatable unit tests, testing can be done as part of the build, ensuring that all features are working correctly each time the build is produced.

2

And builds should be regular, at least daily, so integration is done as you go too.

3

The purpose of these principles is to keep the software in releasable condition throughout the development, so it can be shipped whenever it's appropriate.

Scrum

Overview of the Scrum Practice Framework

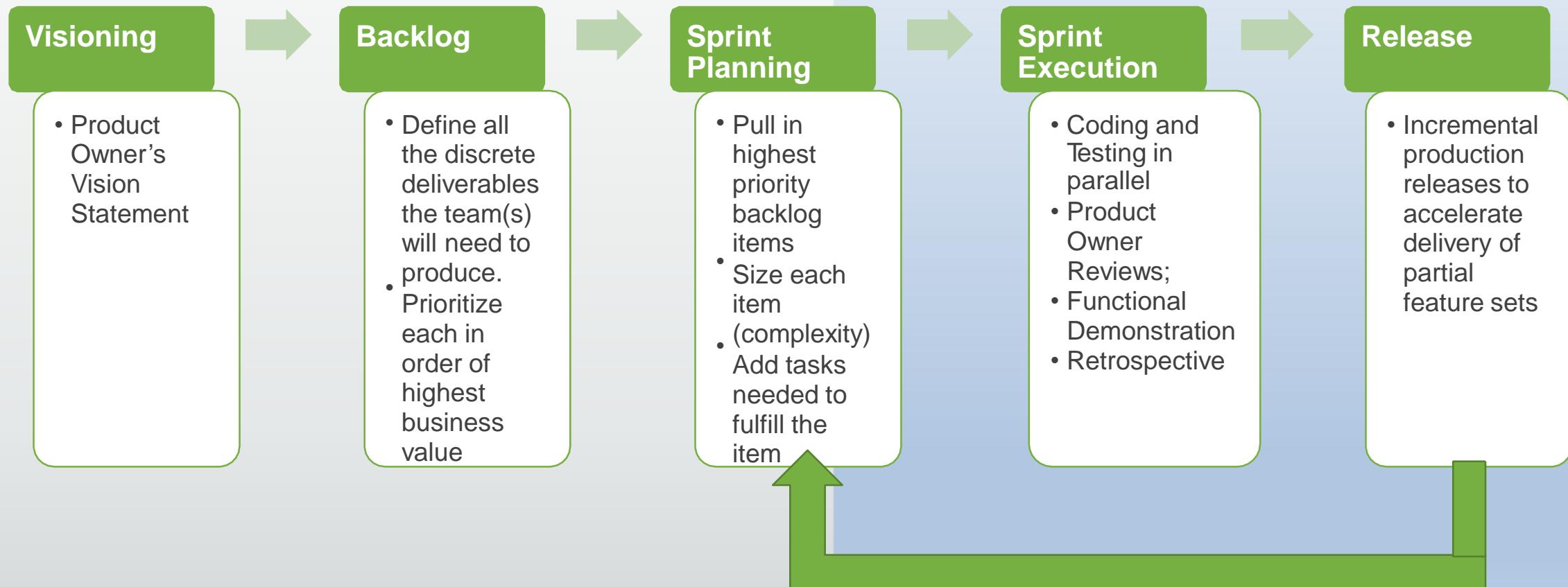
Scrum

Scrum is the framework that helps teams work together.

Gets its name from a Rugby term used as a metaphor to reflect the degree of team cooperation needed to advance the football across the goal line.

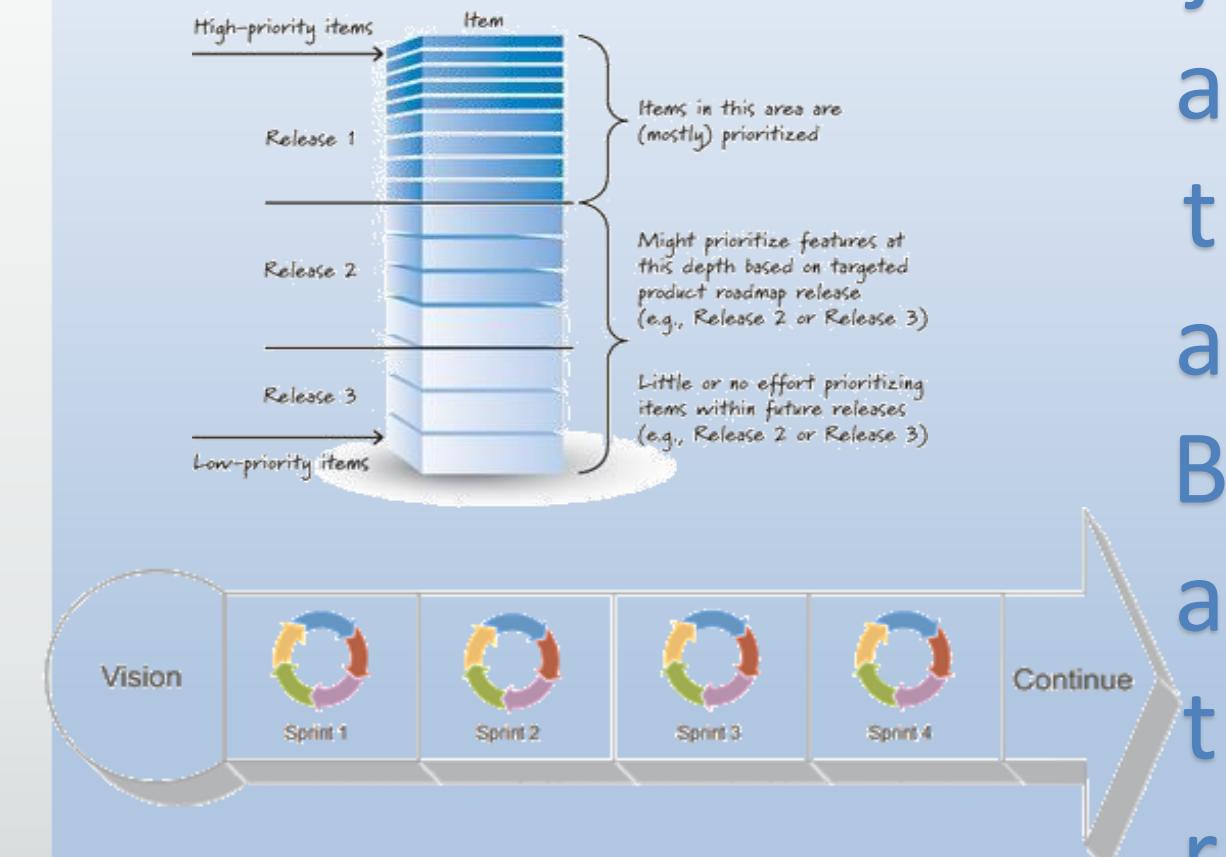


How Does It Work



How Does It Work

- Product Backlog: Single Source of Functional and non-functional requirements.
- Chops up the Product Backlog into a series of smaller pieces
- Each piece is worked within a time boxed period called a Sprint.
- Work is inspected, accepted or rejected each Sprint by the Product Owner (business owner).



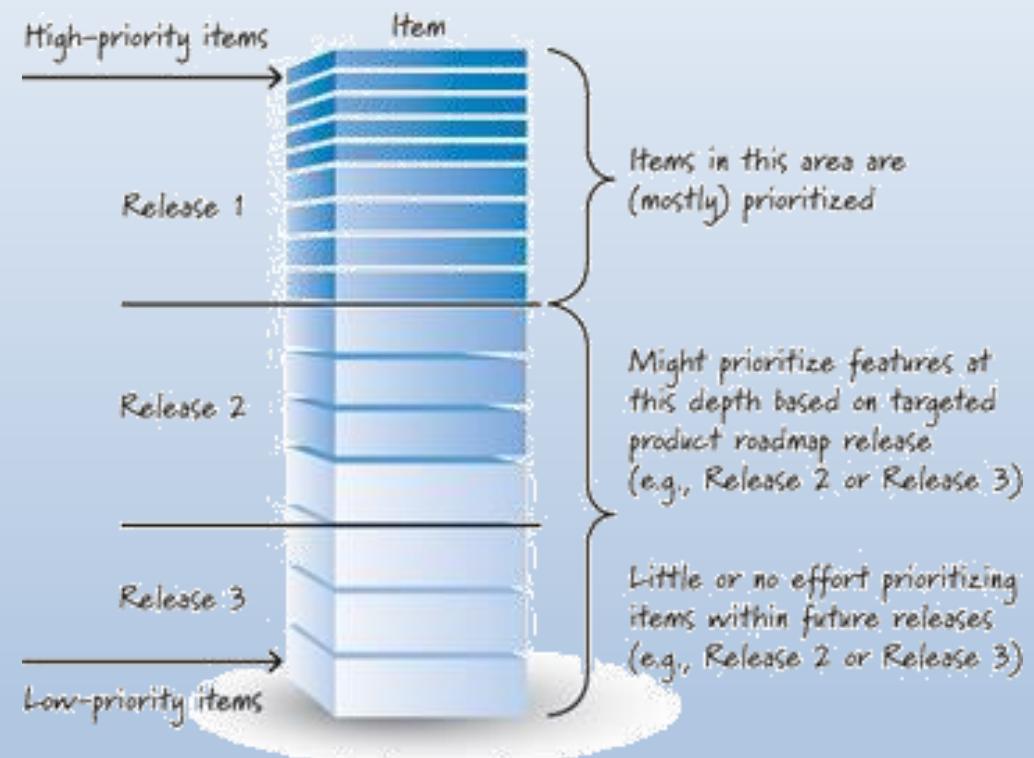
How Does It Work

- **Business Value**

- Work is prioritized highest business value to lower business value.
- Highest value items should be elaborated in detail; ready for the next Sprint Planning.

- **Tactics**

- MoSCoW (must have, should have, could have, won't have)
- REIO (Required, Essential, Important, Optional)
- Cost – Benefit Matrix



User Story

Describes a small discrete “need” from the perspective of a role or persona.



As a [call center agent] (WHO)

I need to [login with my password] (WHAT)

So that [I can access the customer’s reservation to cancel it] (WHY)

Contains acceptance criteria that defines “done” (story is done when . . .)

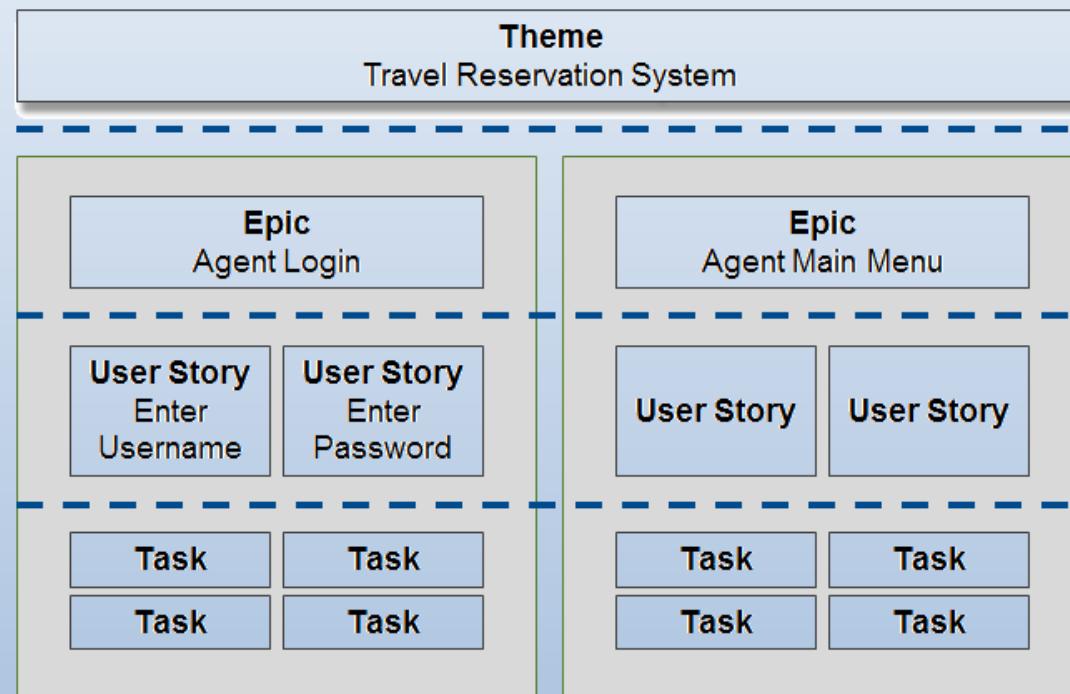
- a premium member can cancel same day without a fee
- a non-premium member is charged 10% for a same-day cancellation
- email confirmation is sent to the customer
- hotel is notified of the cancellation

User Story

- Contains tasks that describe the actions and estimated effort required to fulfill the Story need.
 - Typically starts with a verb, concise, and self evident what the action is and an estimate of effort
 - Create User Table (1 hr)
 - Create password encryption service (4 hr)
 - Create login service (4 hr)
- Is testable (functionally)
 - Well constructed acceptance criteria doubles as functional test criteria for the story (positive and negative)
 - User can login using a valid password
 - User cannot login using an invalid password

User Story Scope

- Theme
 - Very broad high level category of related Epics and Stories
- Epic
 - High level User Story; typically representing a broad functional feature
 - Epics are sometimes referred to as Feature
- User Story
 - Represents a discreet detailed functional requirement.



Story Map

- Make visible the workflow or value chain
- Show relationships of larger stories to child stories
- Help confirm the completeness of the Backlog
- Provide a useful context for prioritization
- Plan releases in complete slices of functionality



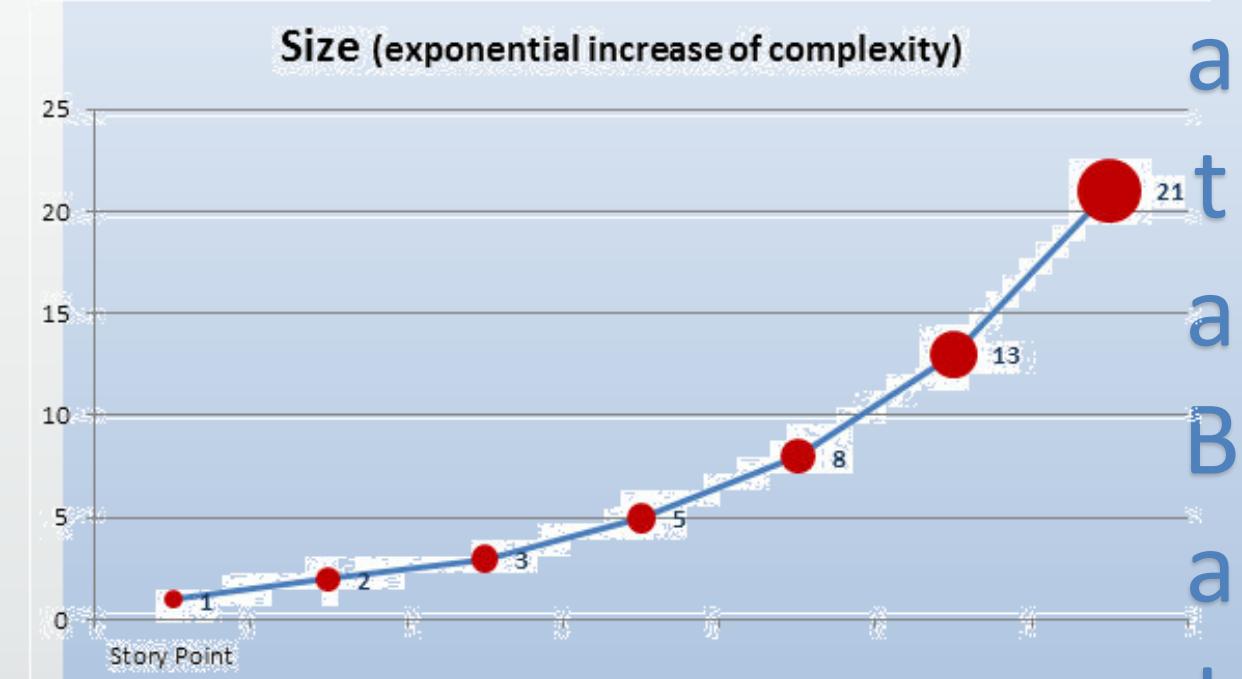
Release Roadmap

- Helps align stakeholder expectations
- List the Release Name or Version Number
- List the goals for each release
- List the Preliminary feature set for each release
- Optionally include metrics that help define if the goal(s) were met

Timeline	2016Q1	2016Q2	2016Q3	2016Q4
Rel ID	R1	R2	R3	R4
Goal	One UI; all admin systems, basic search functions	Add additional search types	Integration of IVR pop, SWAP, and CLASS	Additional Notes Functionality
Features	<ul style="list-style-type: none">• Name search• Organization search• Policy number search• View Contract details (Summary)• Search usage reporting	<ul style="list-style-type: none">• Customer search using last 4 of SSN• Search using FULL SSN• Adjustments to Agent Result Data• Search usage reporting adjustments	<ul style="list-style-type: none">• IVR Pop integration• View note by Policy Number and Owner• SWAP Integration• CLASS Integration	<ul style="list-style-type: none">• Attention and Alert note handling• Copy/paste functionality• Ability to enter notes on UI and write back to source system

Estimating

- Story Points
 - Variation of tee-shirt sizing estimated in points relative to perceived complexity of the story (effort, complexity, and risk)
 - Much quicker and accurate than time spent ‘breaking down and measuring’
- Techniques
 - Planning poker cards
 - Reference Story. 2 story points = ‘small’, size other User Stories relative to that; smaller, larger, same.



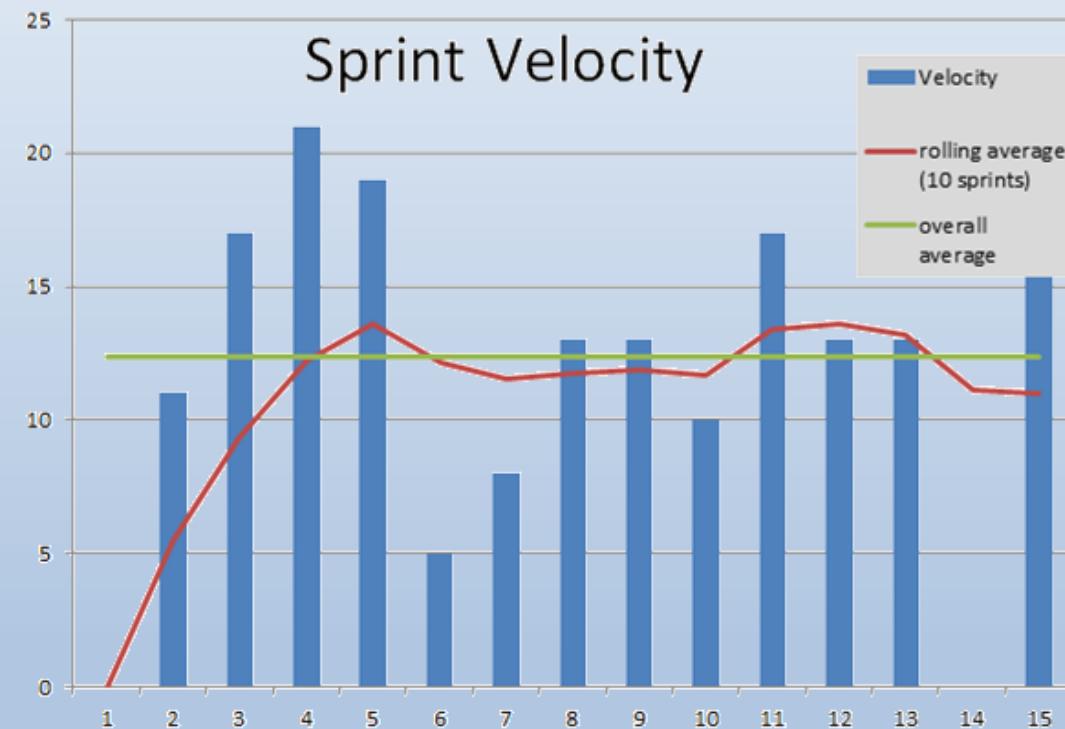
Relative Estimation Advantage

- Humans are terrible at absolute estimation but quite good at relative estimation.
- It is generally faster
- It gets a team thinking (and talking) as a group, rather than as individuals
- It encourages spending analysis time appropriately
- It is cost-effective

Animal	Estimate the weight in pounds	Estimate the weight lightest (1) to Heaviest (5)
Tiger	?	4
Rabbit	?	2
Squirrel	?	1
Elephant	?	5
Impala	?	3

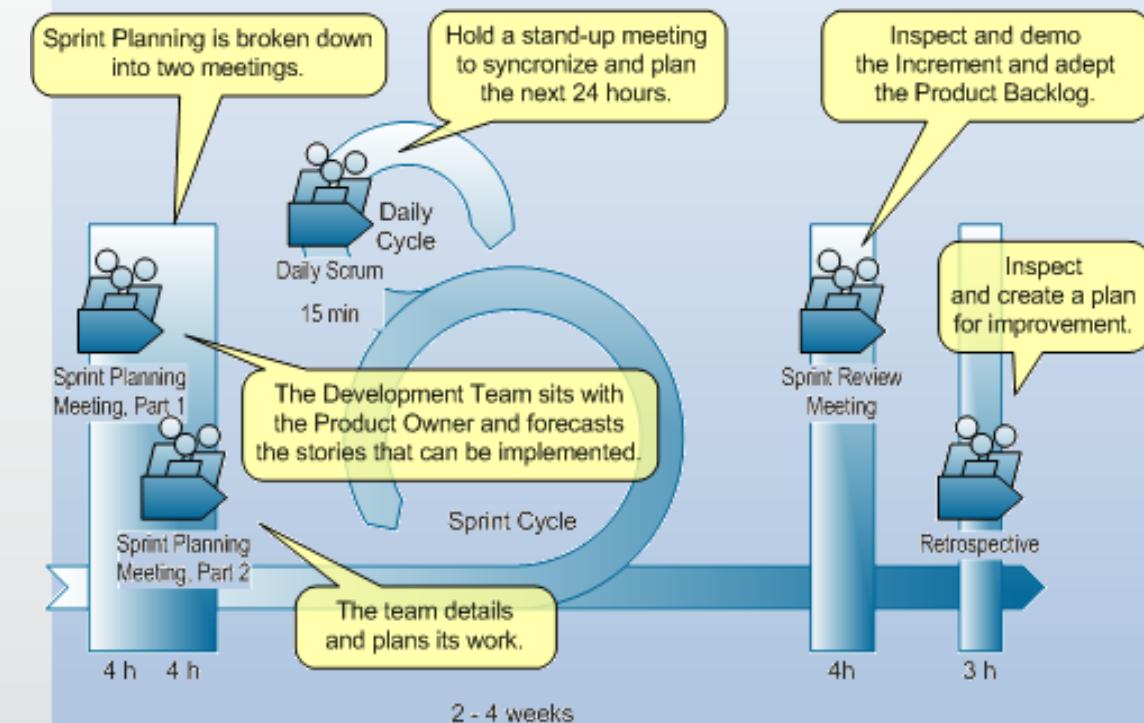
Velocity

- Points total from all completed stories is the team's velocity for that Sprint.
- After several Sprints, velocity "norms". Average velocity then becomes a predictor of Sprint throughput.
- The team can periodically compute estimated project completion based on backlog remaining points



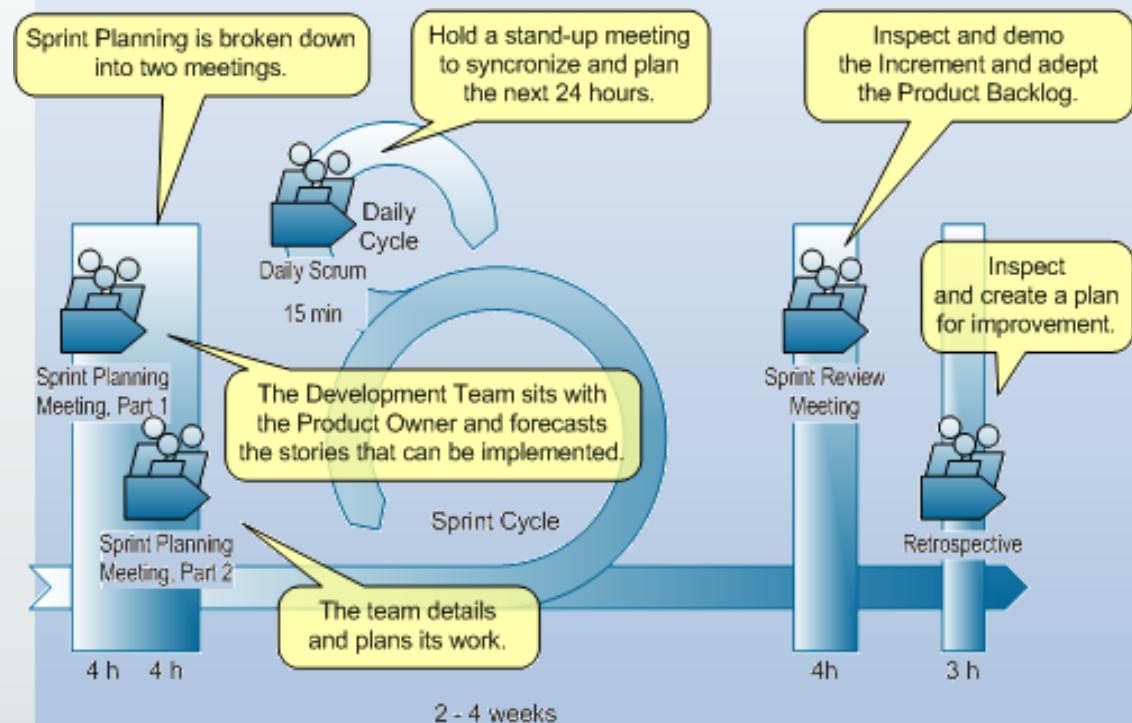
Sprints

- Time boxed
 - Typically 2-4 weeks
- Sprint Planning (Day 1)
 - Pull in the next highest priority items from the backlog.
 - First session with the Product Owner
 - Second session to work out the technical strategy for completing the work.



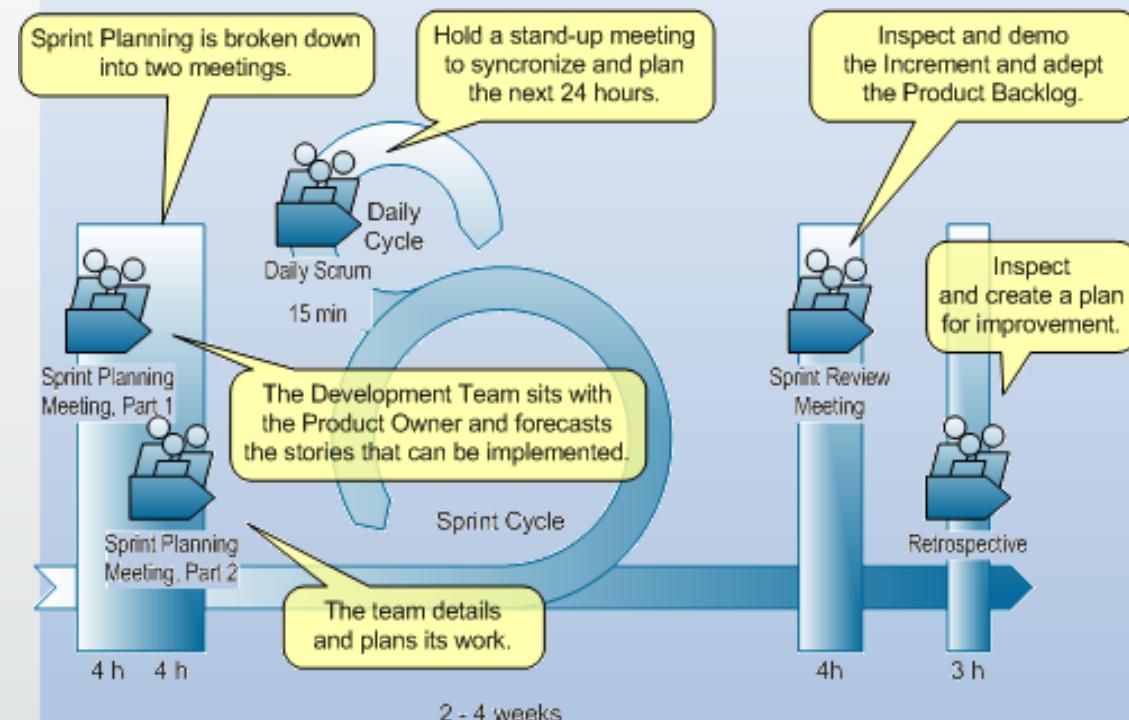
Sprints

- Daily Stand-up (Each Day)
 - Each team member:
 - What they did yesterday
 - What they plan to do today
 - Any impediments blocking progress.
- Sprint Review (Final Day)
 - Product Owner reviews achievements of the Sprint with the team
 - Product Owner and team presents a demonstration or discusses latest functionality with external audience.



How Does It Work

- **Retrospective**
 - The Retrospective, or ‘Retro’, is attended by the Scrum Master and the team and is the final team meeting in the Sprint.
 - The primary purpose is to determine what went well, what didn't go well, and how the team can improve in the next Sprint.
 - The Retrospective is the opportunity for the team to focus on its overall performance and identify strategies for continuous improvement on its processes



Roles



Product Owner

- Represents the Business
- Defines requirements (the backlog)
- Accepts or rejects team output
- Makes business decisions
- Provides visibility to leadership



Scrum Master

- Scrum process expert
- Ensures consistent team practices
- Coaches team and individuals; to maximize efficiency and quality
- Partners with the Product Owner to maximize alignment
- Assists with logistics, admin, or impediment removal to ensure team can run full throttle.



IT Team

- Typically 7 +/- 2 members
- Armed with skills to deliver increments of working software
- The team is empowered to organize/execute work and to solve problems within their control
- Cross-functional; members learn a bit of how other work is done so they can assist as needed.

Scrum Master

- Duties and Allocations
 - People: Gate keeper; shield the team from undue interruptions and distractions, build and maintain communication between the team and everybody else external to the team.
 - Process: Scrum process activities and meetings.
 - Delivery: Ongoing backlog refinement sessions, impediment management, delivery coordination and status meetings, governance / PMO administrative tasks.

Scrum Master Duties and Time Allocations (approximate)	2 Week (10 day) Sprint	3 Week (15 day) Sprint
Gross Capacity	80 Hours	120 Hours
People	10 13%	15 13%
Gatekeeper: Interface point between team and management or stakeholders. Shield the team from undue interruptions.	10	15
Relationships management; help build and maintain communication and trust within the team and between the team and everybody else external to the team.		
Process	19 24%	22 18%
Daily SCRUM Meetings	5	8
Sprint Planning Meeting	8	8
Sprint Review Meeting	3	3
Sprint Retrospective Meetings	3	3
Delivery	36 45%	54 45%
Ongoing Backlog Refinement	12	18
Impediment Management	10	15
Delivery coordination and status meetings	10	15
Governance / PMO administrative tasks	4	6
Uncommitted Hours	15 19%	29 24%
Utilization	81%	76%

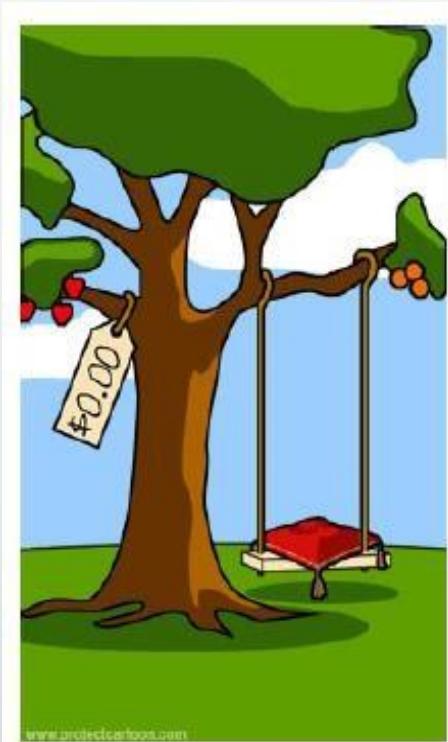
Business Analyst

- Assists the Product Owner and the Team
 - The Product Owner has a full time job
 - The Product Owner defines the high level functional deliverables (Epics) and priority
 - The BA digs out the detail of each high level functional deliverable into users stories
 - The BA helps create minimum needed designs
 - Pre-Validates the Story as “Done”
 - Helps prepare and execute test plans



Putting It All Together

The Customer Needs This

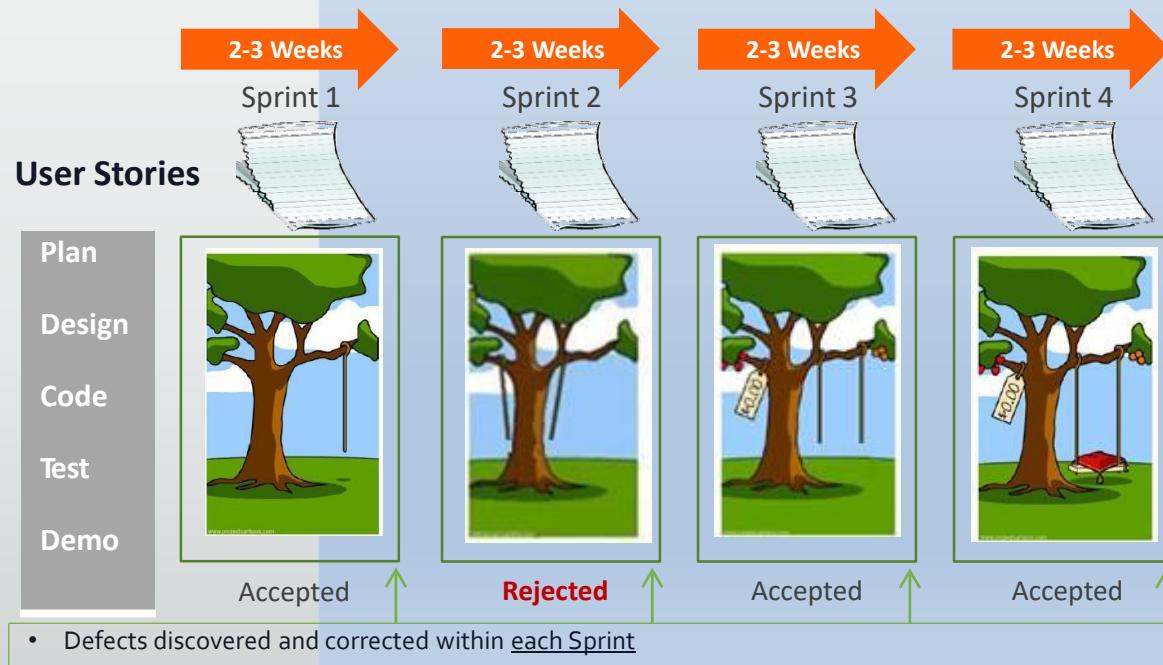


Waterfall

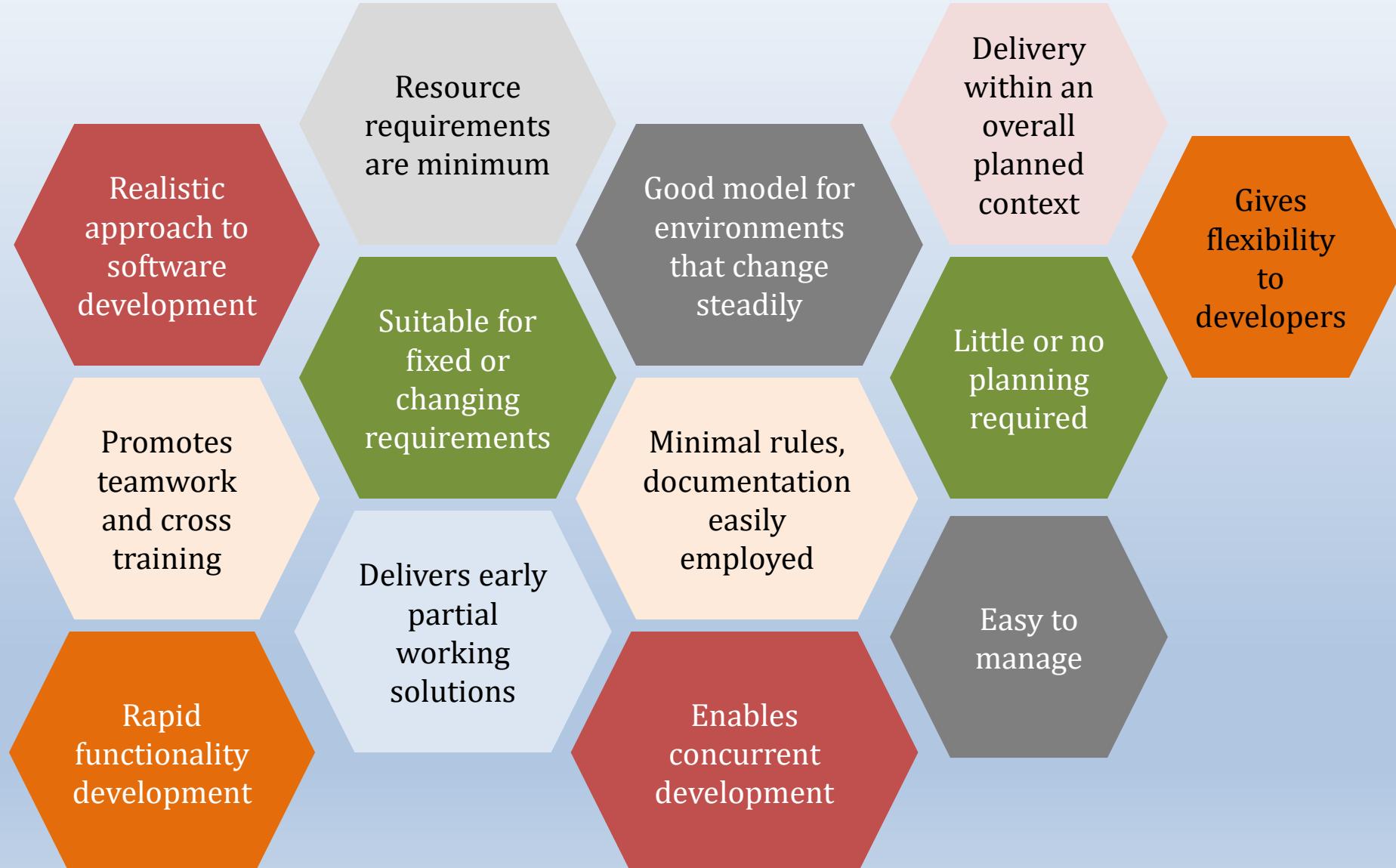


Agile
(Scrum)

1 User Story =
1 Functional
Requirement



Agile Model - Advantages



Agile Model - Disadvantages

1

Not suitable for handling complex dependencies.

2

More risk of sustainability, maintainability and extensibility.

3

Overall plan is a must.

4

Strict delivery management to meet deadlines.

5

Depends heavily on customer interaction.

6

Very high individual dependency.

7

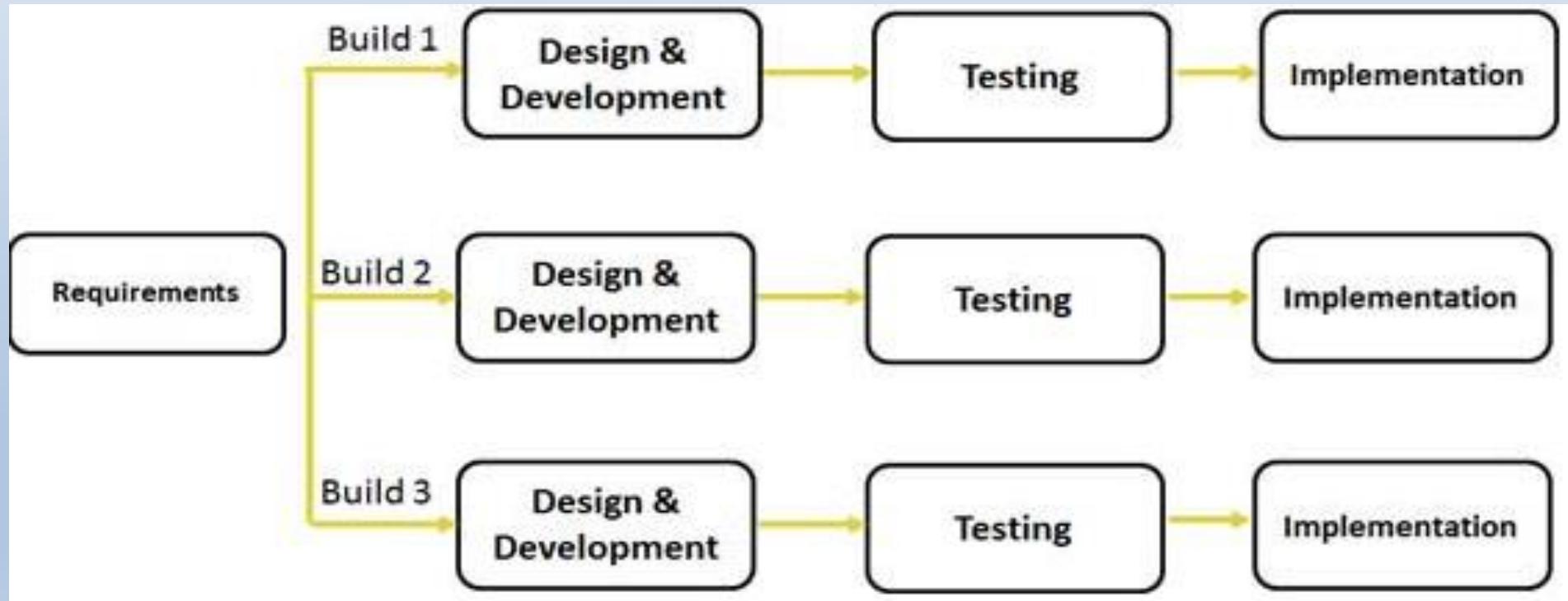
Technology transfer is challenging.

8

Lack of documentation.

Iterative or Incremental Project Life Cycle

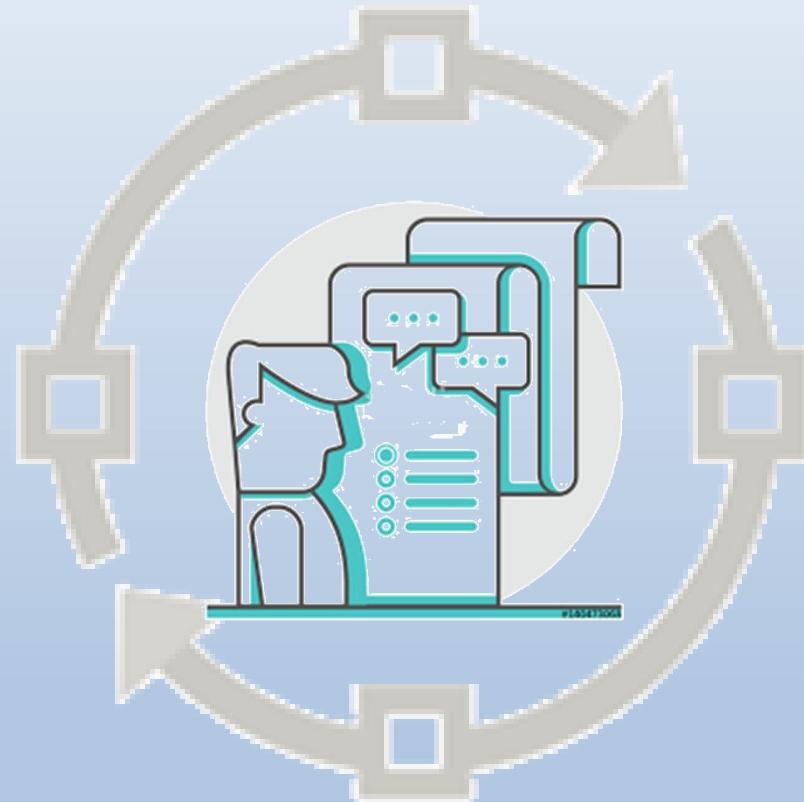
In an iterative model, the iterative process starts with a simple implementation of a small set of software requirements and iteratively enhances the evolving versions. This cycle is repeated until the complete system is implemented and ready to be deployed.



Introduction to Iterative Project Life Cycle

An iterative life cycle model does not attempt to start with a full specification of requirements.

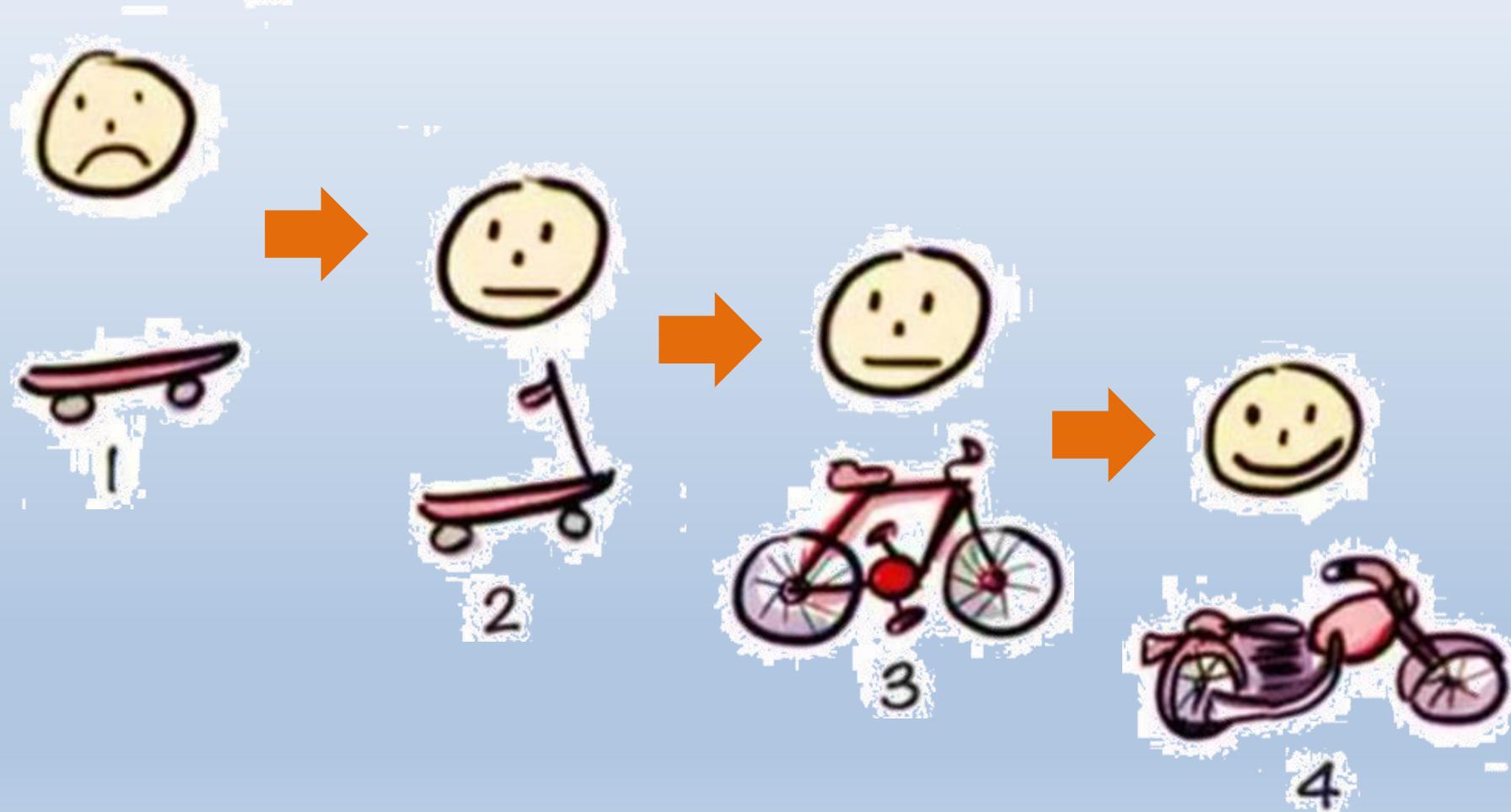
Development begins by specifying and implementing just part of the software, which can then be reviewed in order to identify further requirements.



This process is then repeated, producing a new version of the software for each cycle of the model.

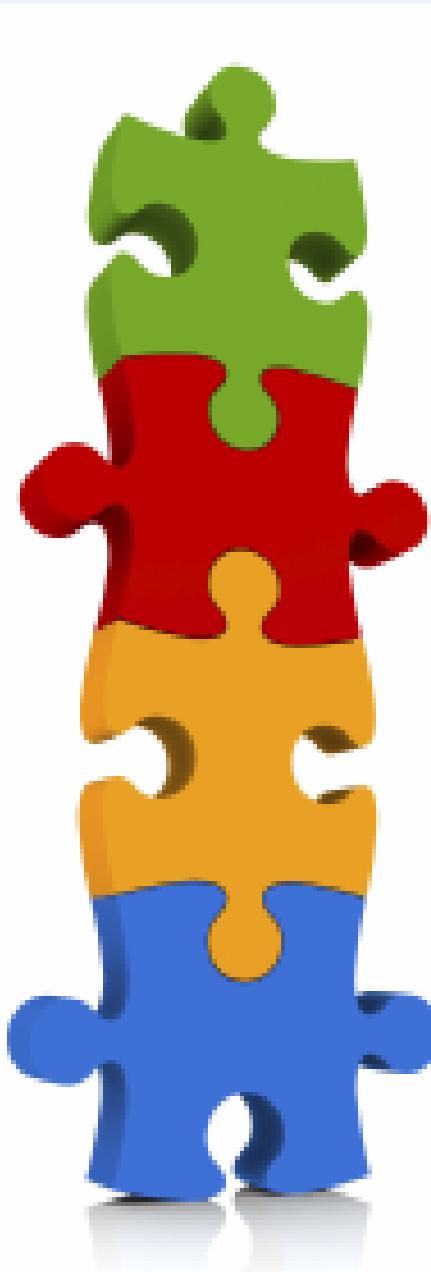
Iterative Model

Iterative work creates a rough product or piece in one iteration. This product or piece is then reviewed and will be improved in next iteration and so on until it's finished.



Hence, in an iterative model the whole product is developed step by step.

Iterative Model Application

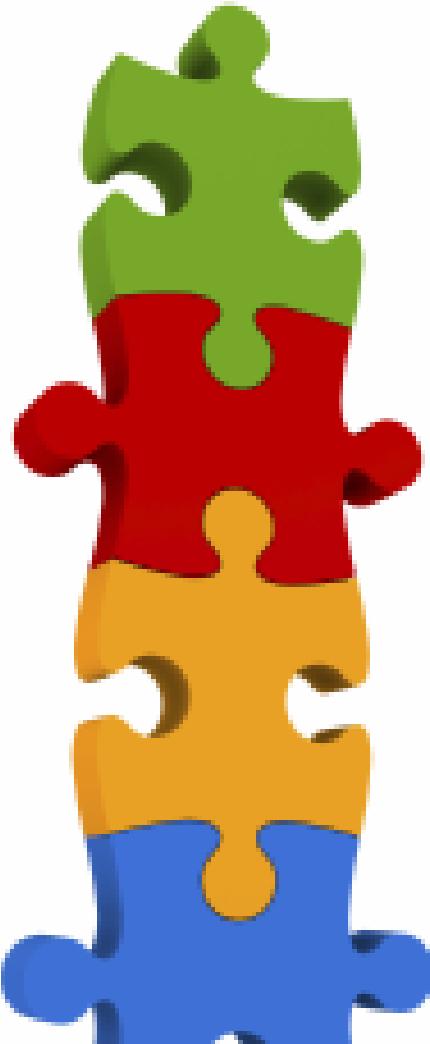


Requirements of the complete system are clearly defined and understood.

Major requirements must be defined; however, some functionalities or requested enhancements may evolve with time.

There is a time to the market constraint.

Iterative Model Application



A new technology is being used and is being learnt by the development team while working on the project.

Resources with needed skill set are not available and are planned to be used on contract basis for specific iterations.

There are some high risk features and goals which may change in the future.

As the model name suggests, the evolution takes its time and pace as acceptable to project!

Iterative Model - Advantages



01

An iterative model creates a high-level design of the application before building the product and defining the design solution for the entire product.

02

Later, design and built a skeleton version, and then evolve the design based on what had been built.

03

An iterative model builds and improves the product step by step.

04

Hence, defects can be tracked at early stages.

Iterative Model - Advantages



05

This avoids the downward flow of the defects. Testing and debugging during smaller iteration is easy.

06

Reliable user feedback is possible. Risks are identified and resolved during iteration; and each iteration is an easily managed milestone.

07

When presenting sketches and blueprints of the product to users for their feedback, users imagine how the product will work.

08

In an iterative model lesser time is spent on documenting and more time is given for designing.

Iterative Model - Disadvantages



Applicable only to large and bulky software development projects.

Hard to break a small software system into further small serviceable increments/modules.

More resources may be required.

Although cost of change is lesser but it is not very suitable for changing requirements.

Iterative Model - Disadvantages



More management attention is required.

System architecture or design issues may arise because not all requirements are gathered in the beginning of the entire life cycle.

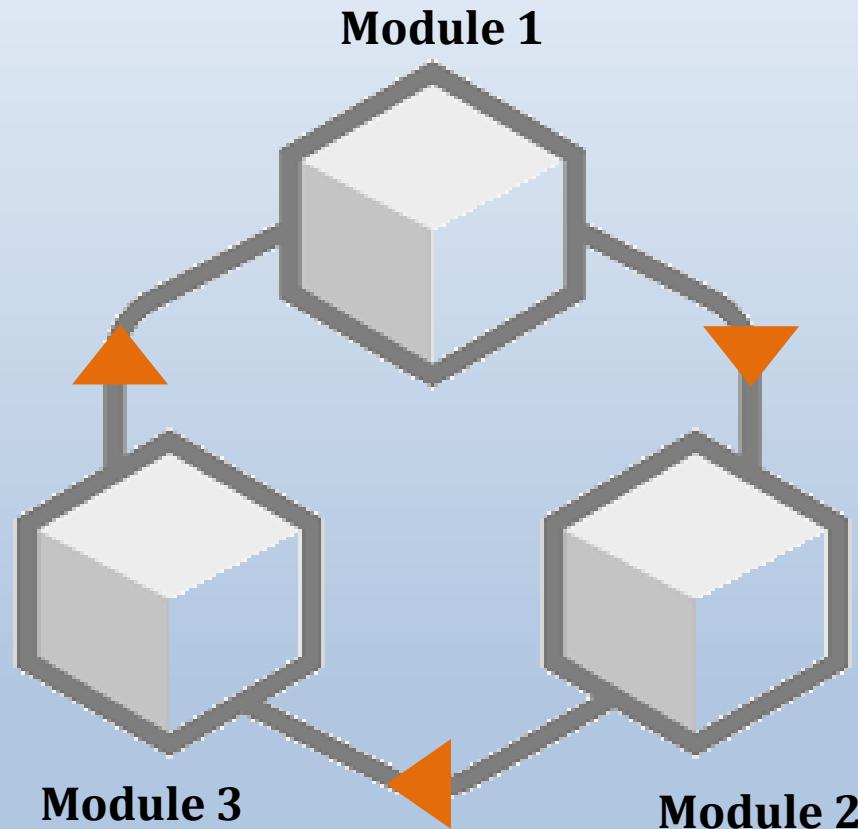
Defining increments may require definition of the complete system

Not suitable for smaller projects

Incremental Model

The whole requirement is divided into various builds.

Multiple development cycles take place here, making the life cycle a “multi-waterfall” cycle.



Cycles are divided up into smaller, more easily managed modules.

Incremental model is a type of software development model like V-model, Agile model etc.

Incremental Model

Each module passes through the requirements, design, implementation and testing phases.

A working version of software is produced during the first module, so you have working software early on during the software life cycle.



Each subsequent release of the module adds function to the previous release.



The process continues till the complete system is achieved.



Incremental Model

When we work incrementally we add piece by piece but expect that each piece is fully finished.

For example:



As in the image above:

01

A person has thought of the application

02

The first iteration the first module of the application or product is ready and can be demoed to the customers.

03

The other module is ready and integrated with the first module

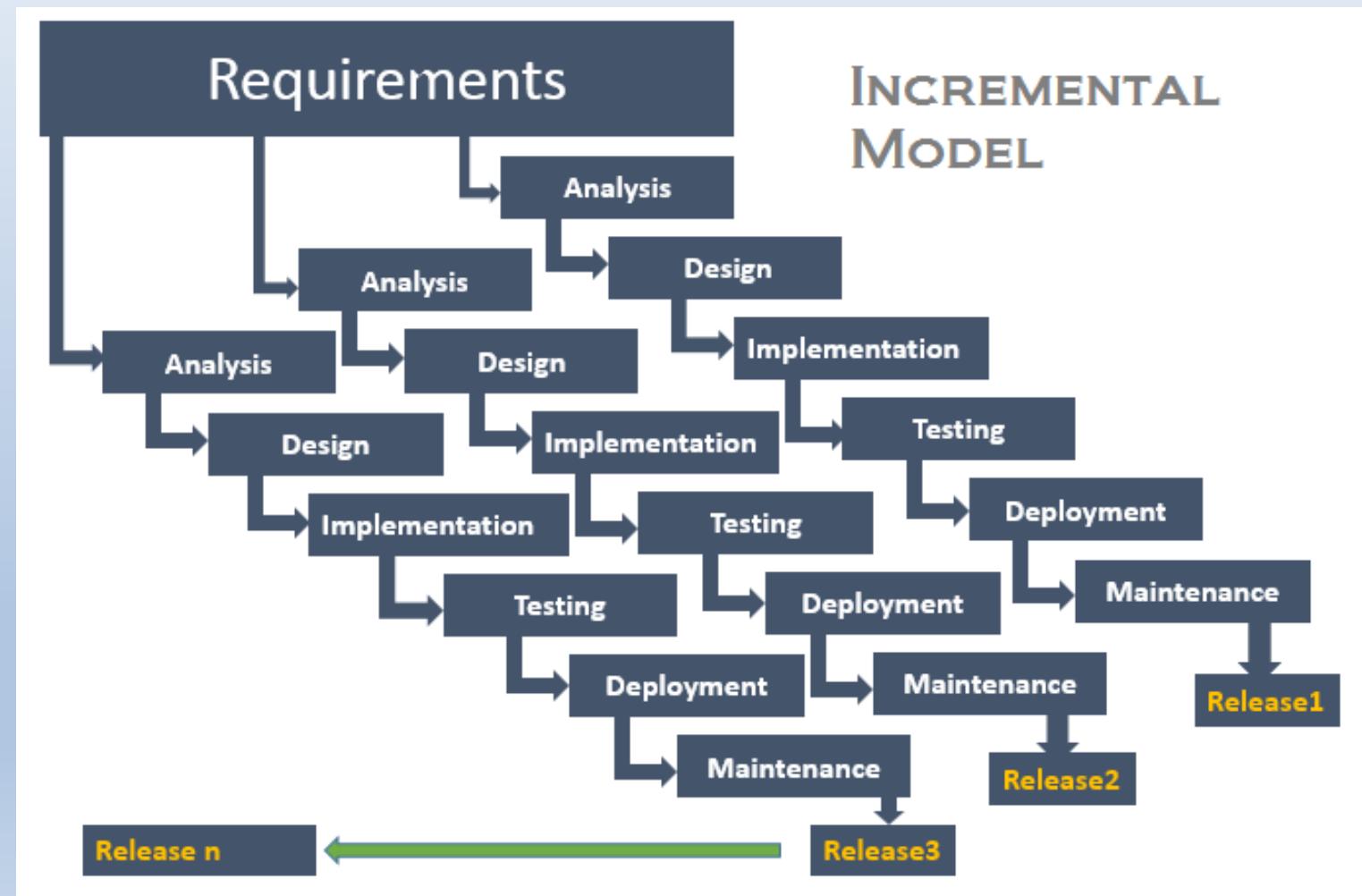
04

The whole product is ready and integrated.

Hence, the product got ready step by step.

Incremental Model Applications

The Incremental model is an evolution of the Waterfall model, where the Waterfall model is incrementally applied.



Incremental Model- Advantages



Generates working software quickly and early during the software life cycle.



This model is more flexible – less costly to change scope and requirements.



It is easier to test and debug during a smaller iteration.



In this model customer can respond to each built.

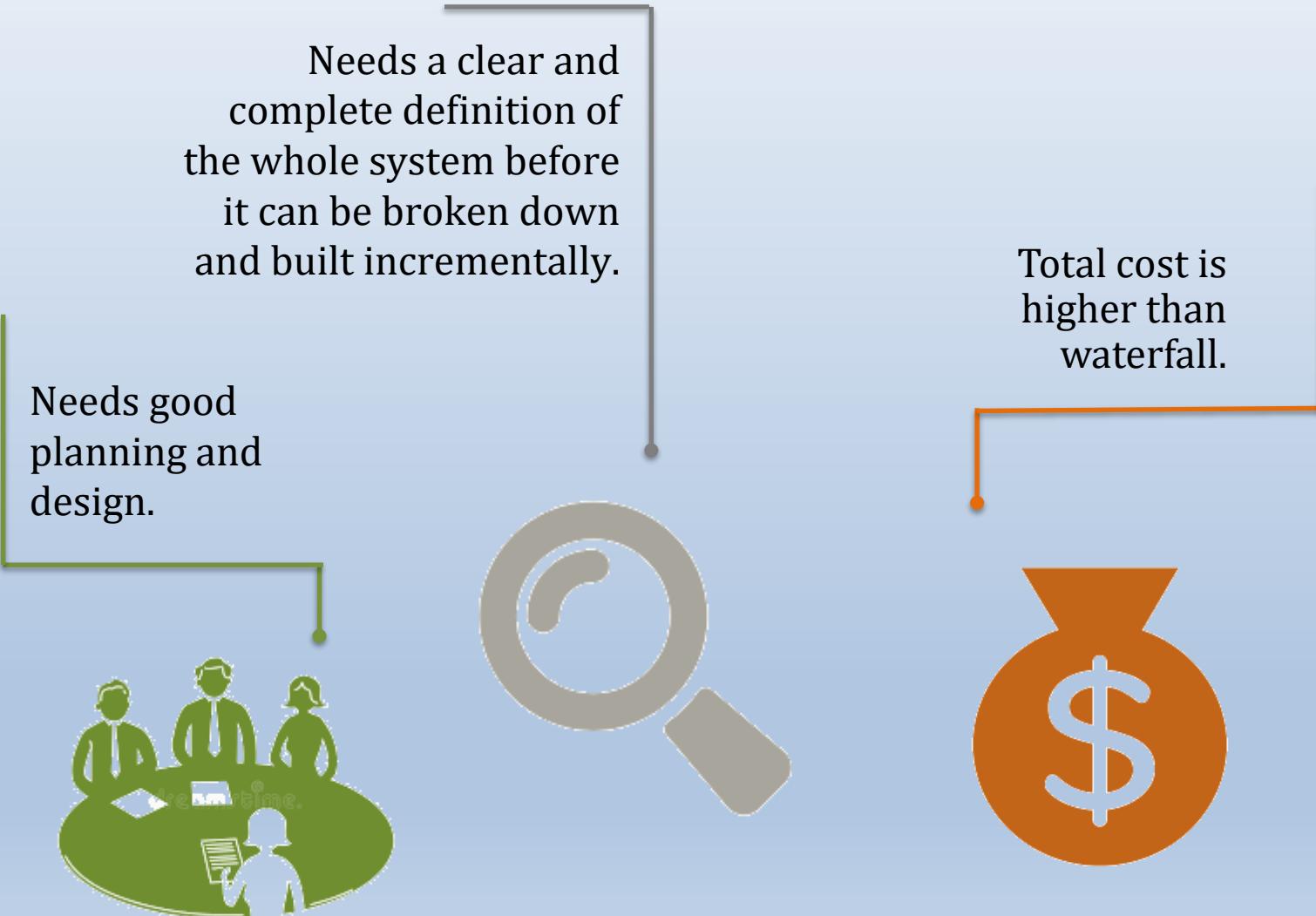


Lowers initial delivery cost.

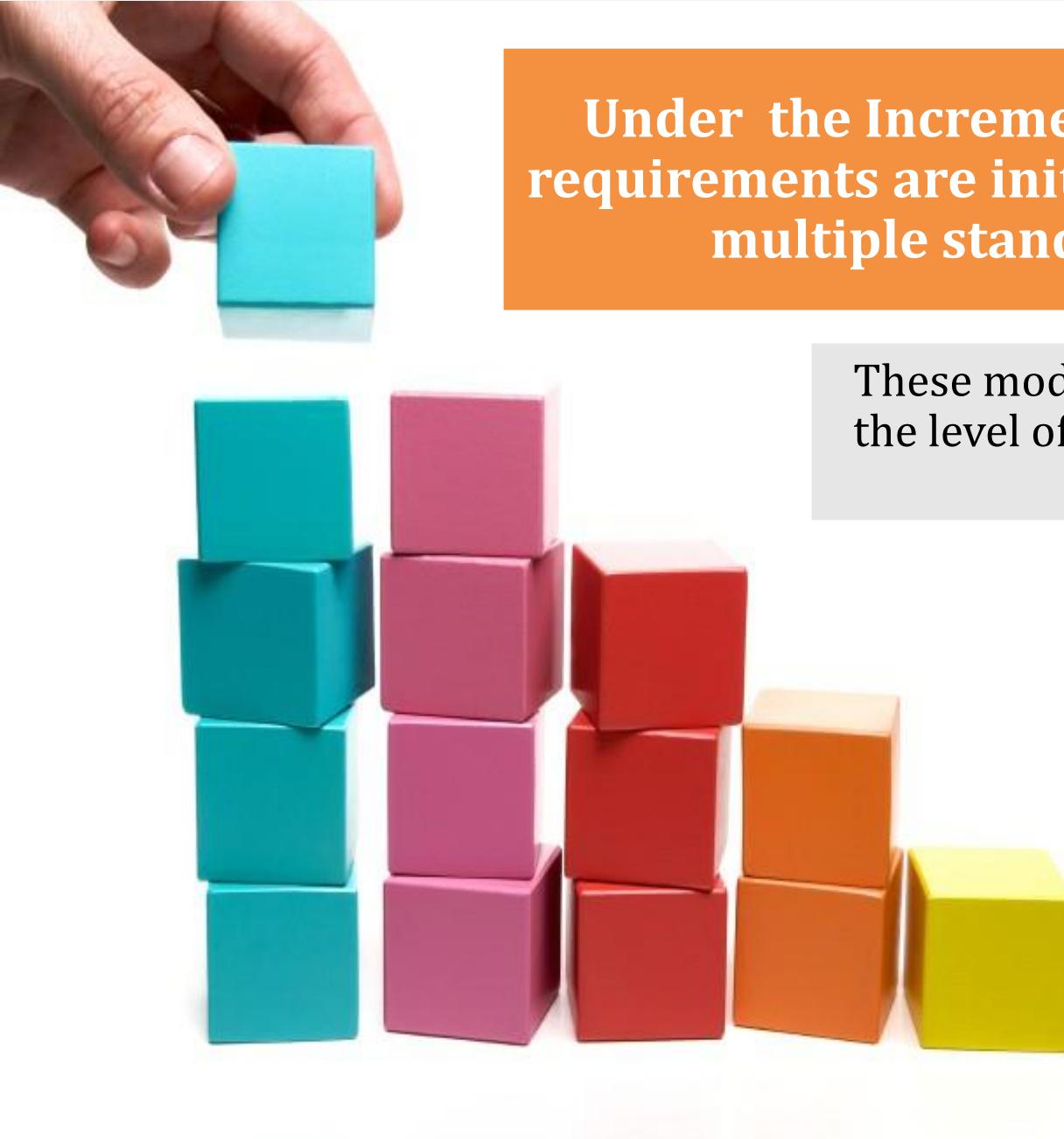


Easier to manage risk because risky pieces are identified and handled during it'd iteration.

Incremental Model- Disadvantages



Key Highlights

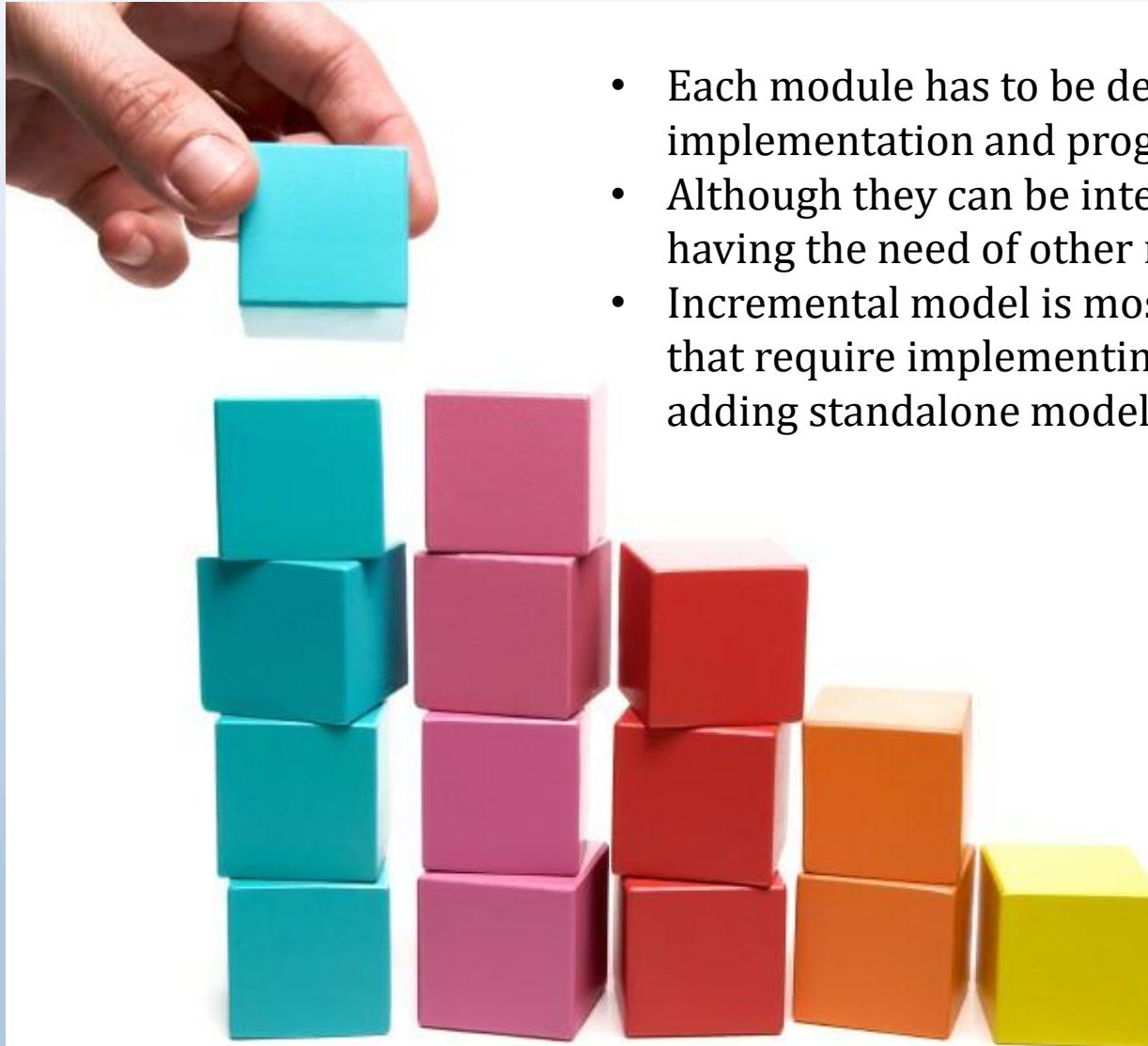


Under the Incremental Model, software requirements are initially broken down into multiple standalone modules.

These modules are drafted according to the level of priority they have under the software project.

Every module is a standalone function

Key Highlights



- Each module has to be developed according to the implementation and progression of the project.
- Although they can be inter-related they can exist without having the need of other modules and functionality.
- Incremental model is mostly followed by large projects that require implementing individual functions and adding standalone models in the long run.

Mostly
followed by
large
projects

A customized step-by-step approach
generally adopts an incremental approach.

Example of Incremental Model



A bank wants to develop software to automate the banking process for insurance services, personal banking, and home and automobile loans.

To enhance the customer services



Example of Incremental Model



You can implement the incremental approach to develop the banking software.

1st Increment

You can implement the personal banking feature and deliver it to the customer.

Later Increments

you can implement the insurance services, home loans, and automobile loans features of the bank.