UNIT-III

Workflows and checking of process.

1) Software process workflows:-

→ the term workflow is used to mean a thread of cohesive and mostly sequential activity.

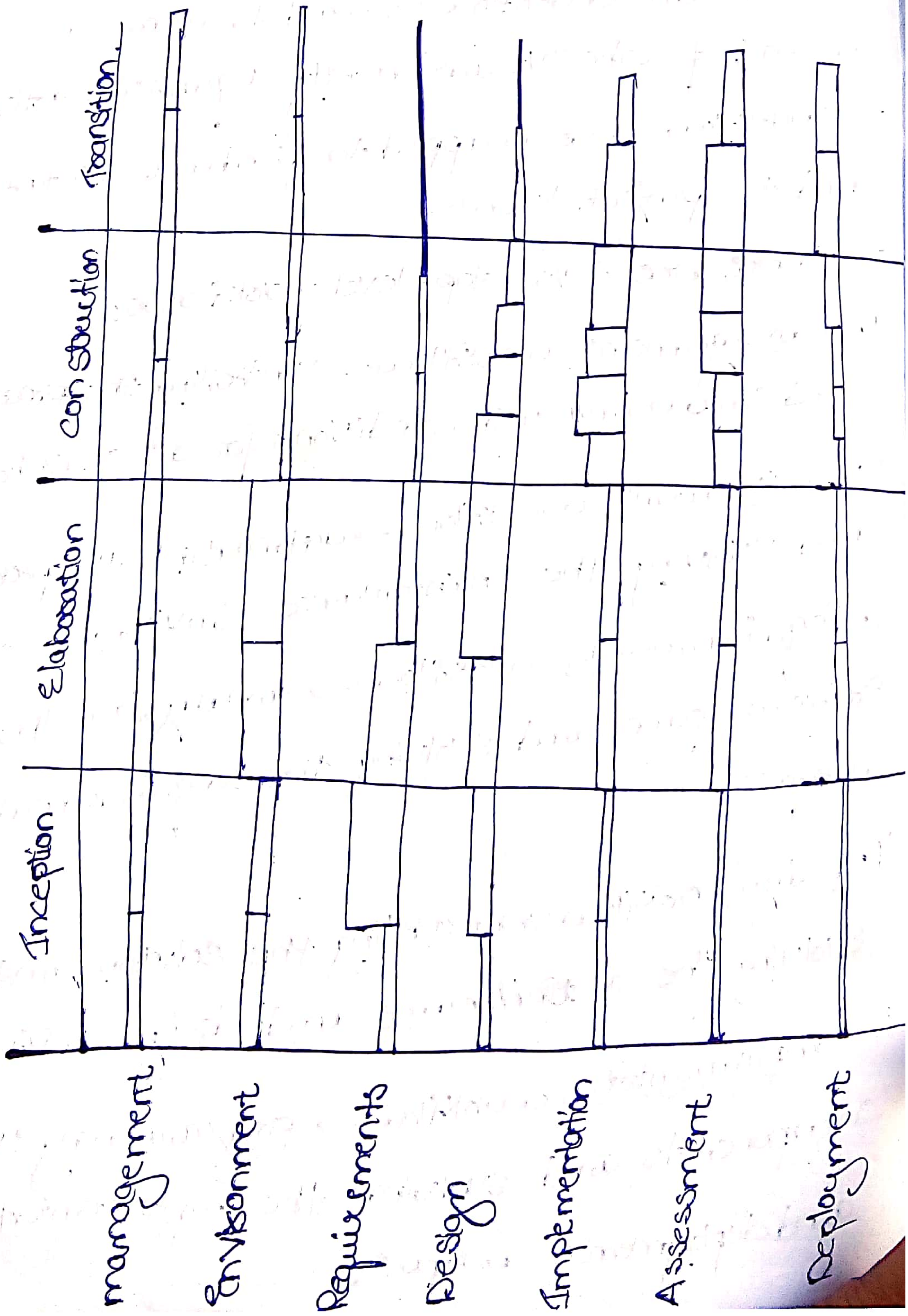
→ workflows are mapped to product artifacts and to project teams.

→ there are seven top-level workflows:

1. management workflow → controlling the process and ensuring win conditions for all stakeholders.
2. Environment workflow → automating the process and evolving the maintenance environment.
3. Requirements workflow → analyzing the problem space and evolving the requirements artifacts.
4. Design workflow → modeling the solution and Evolving the architecture and design artifacts.
5. Implementation workflow → programming the components and evolving the implementation and deployment artifacts.

6. Assessment workflow → assessing the trends in process and product quality.

7. Deployment workflow → transitioning the end products to the user.



1. Architecture-first approach:- Extensive requirements analysis, design, implementation, and assessment activities are performed before the construction phase, when full-scale implementation is the focus.

2. Iterative life-cycle process:- Each phase portrays at least two iterations of each flow. This default is intended to be descriptive, not prescriptive.

→ Some projects may require only one iteration in a phase, others may require several iterations.

3. Round-trip Engineering:- Raising the environment activities to a first-class workflow is critical.

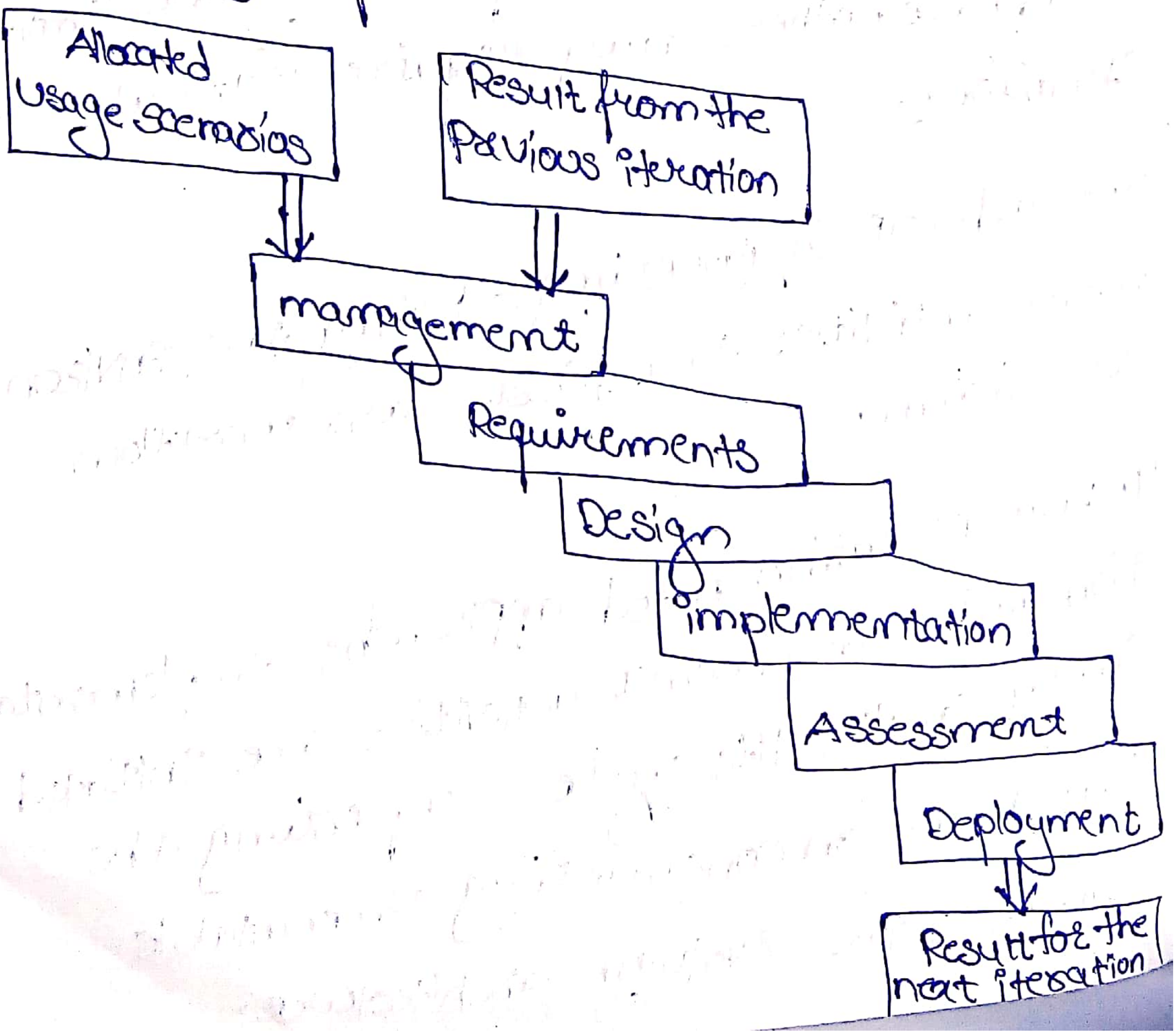
4. Demonstration based approach:- Implementation and assessment activities are initiated early in the life cycle, reflecting the emphasis on constructing executable subsets of the evolving architecture.

2) Iteration workflows:-

→ An iteration consist of loosely sequential set of activities in various proportions, depending on where the iteraction is located in the development cycle.

→ Each iteration is defined in terms of a set of allocated usage scenarios.

→ An individual iteration's workflow, illustrated in below diagram, generally includes the following sequence.


Allocated Usage Scenarios
Result from the Pevious iteration
management
Requirements
Design
implementation
Assessment
Deployment
Result for the next iteration

* **management** :- assignment of work packages, or tasks, to the development team.

* **Environment**:- Evolving the software change order database to reflect all new baselines and change to existing baselines for all products, test, and environment components.

* **Requirements**:- updating any requirement set artifacts to reflect changes necessitated by results of this iteration's engineering activities.

* **Design**:- updating design set artifacts to reflect changes necessitated by results of this iteration's Engineering activities.

* **Implementation**:- integrating and testing all new and modified components with Existing baselines.

* **Assessment**:- assessing results to improve the basis of the subsequent iteration's plan.

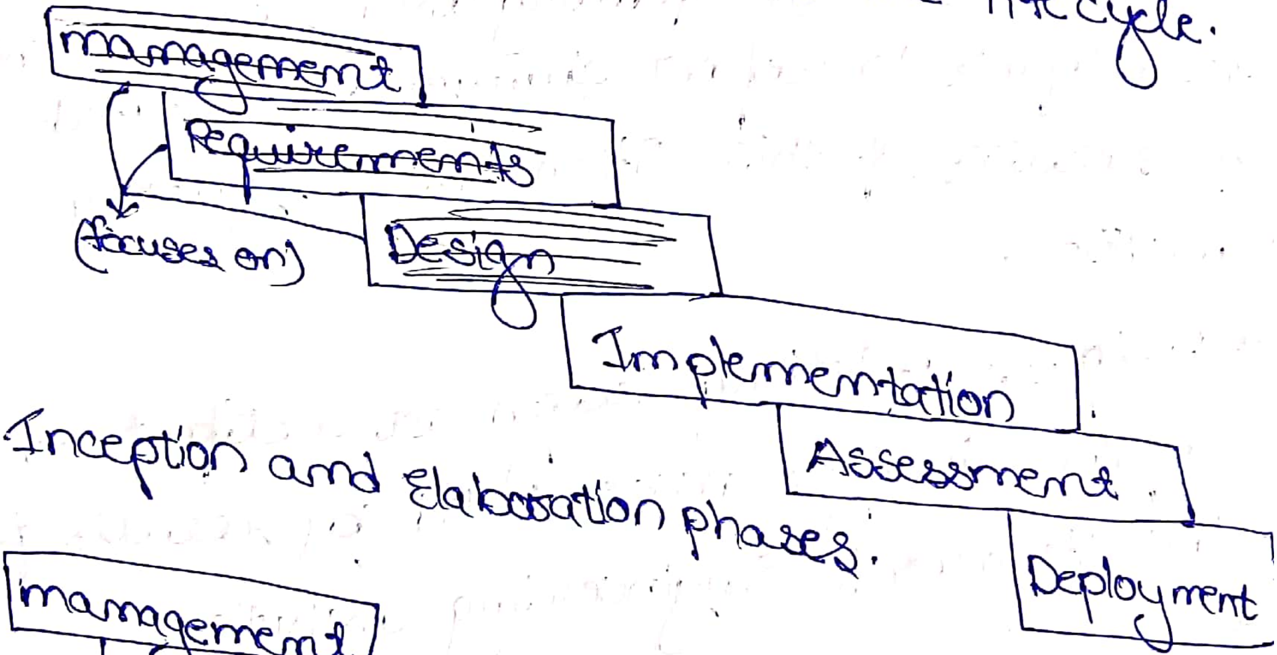
* **Deployment**:- transitioning the release Either to an external organization or to a post-

by conducting

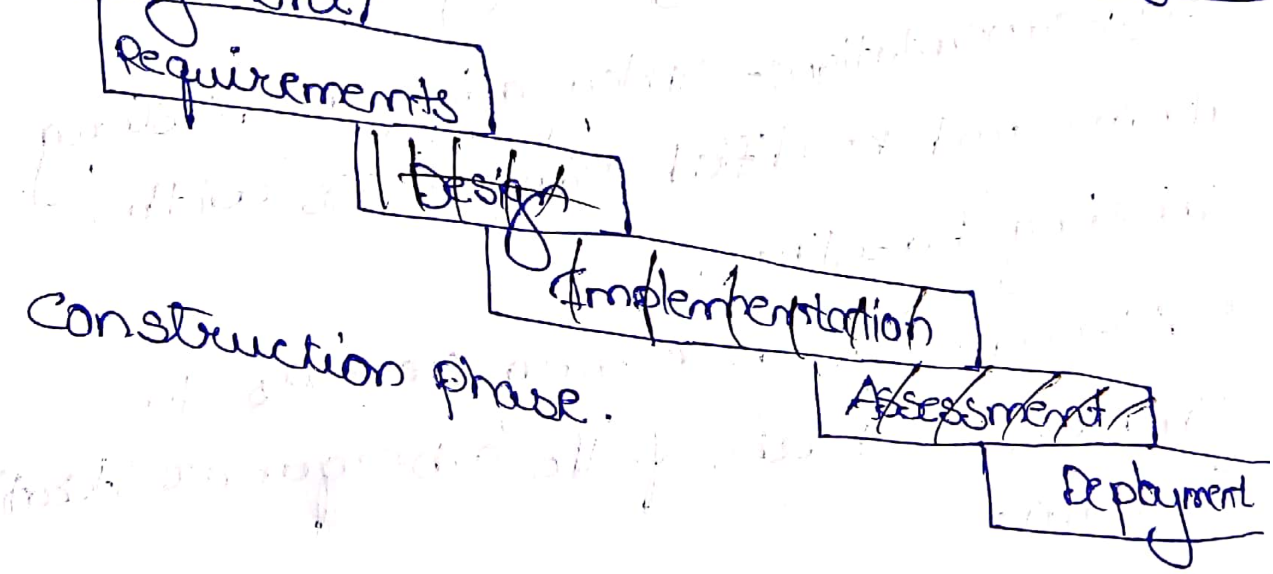
mechanism so that lessons learned can be captured and reflected in the next iteration.

→ As with any sequence of a software development workflow, many of the activities occur concurrently.

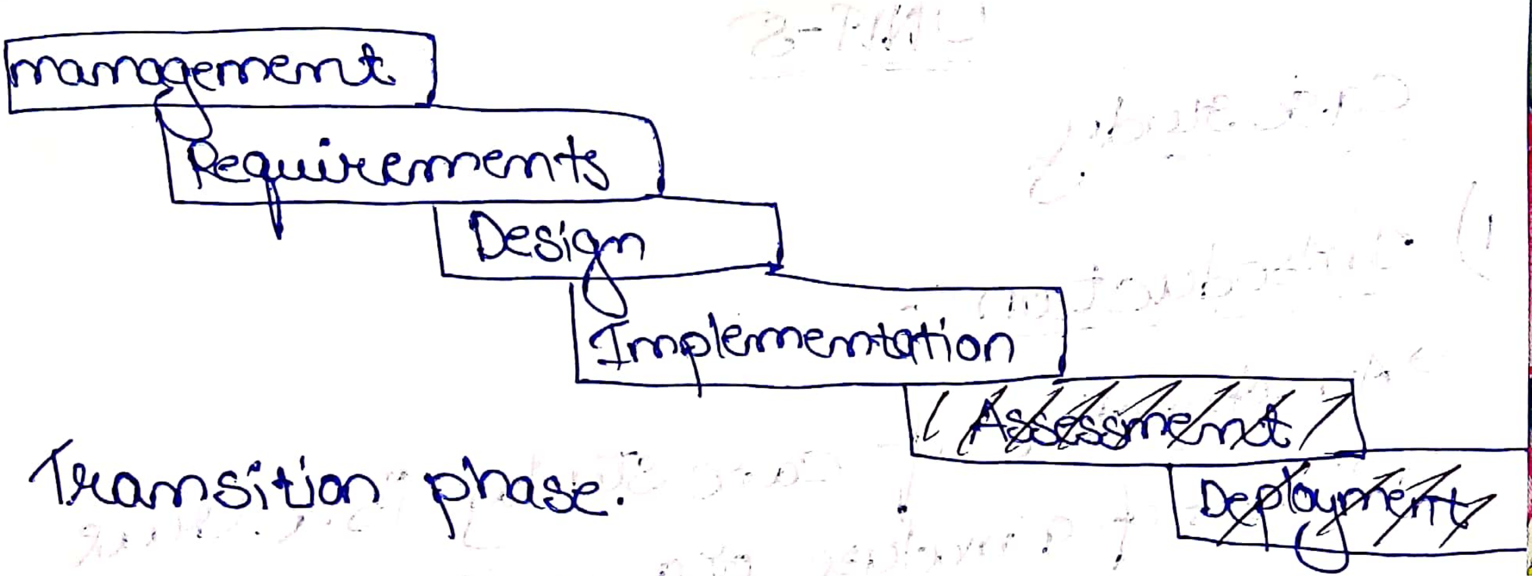
→ below figure shows the emphasis on different activities across the lifecycle.



Inception and Elaboration phases.



Construction phase.



Transition phase.

UNIT-3

1) Minor milestones :-

→ the number of iteration-specific, informal milestones needed depends on the content and length of the iteration.

→ Iterations which have one-month to six-months duration have only two milestones are needed.

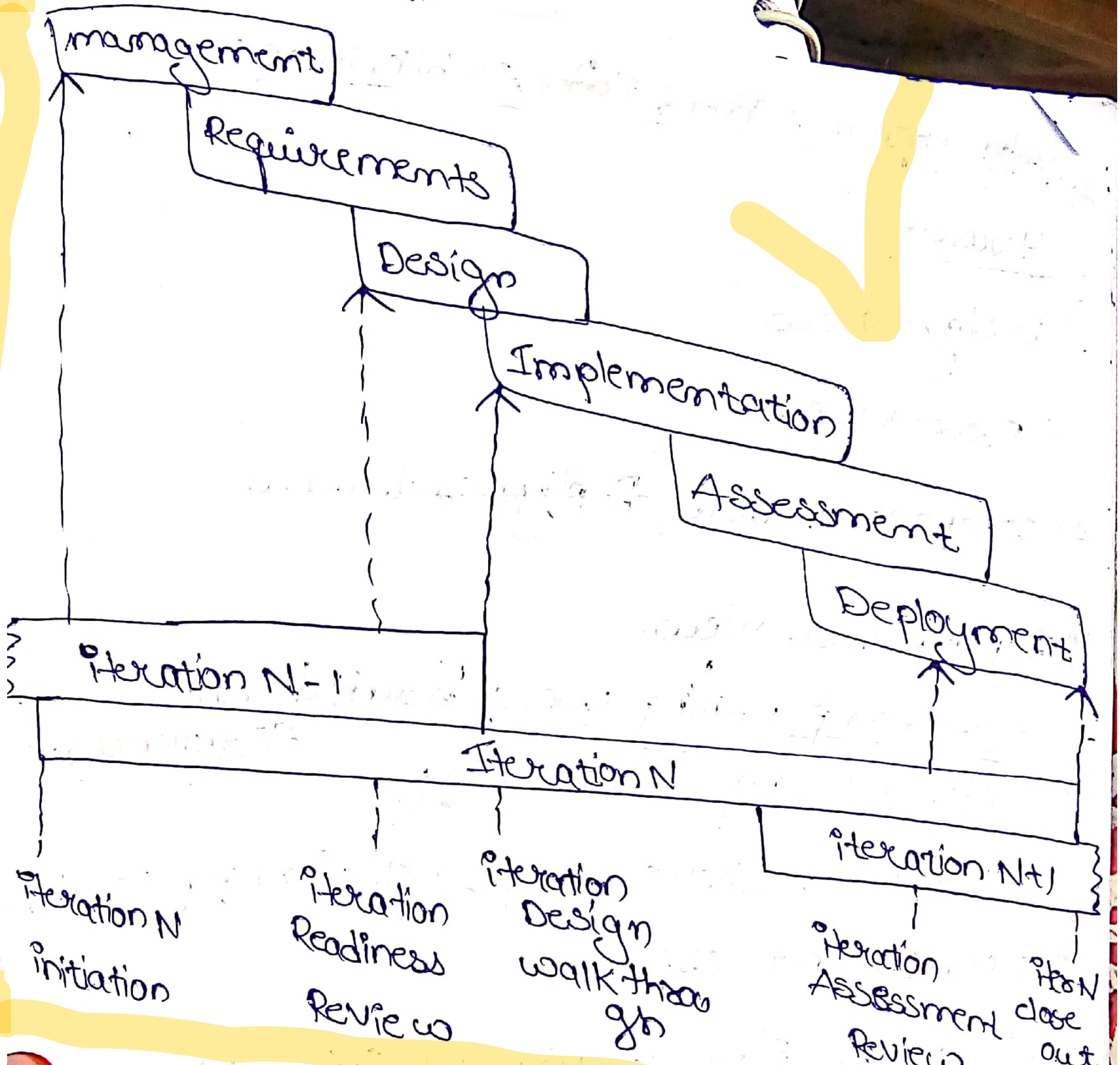
→ for longer iterations some other intermediate review points are added.

→ All iterations are not created equal.

→ An iteration take diff forms and priorities, depending on where the project is in the life cycle.

→ Early iterations focus on analysis and design.

→ later iterations focus on completeness, consistency, usability and change management.



Typical minor milestones in the life cycle of an iteration.

Iteration Readiness Review:- this informal milestone is conducted at the start of each iteration to review the detailed iteration plan, and the evaluation criteria that have been allocated to this iteration.

Iteration Assessment Review:- this informal milestone is conducted at the end of each iteration to assess the degree.

~~4 views~~

~~1) Design view~~

~~2) Process view~~

~~3) component view/~~ ~~Deployment view~~

~~4) Deployment view.~~

~~process planning work break down~~

2) iterative process planning:- structures:-

→ like s/w development, project planning is also an iterative process.

→ like s/w, plan is also an intangible one.

→ plans have an engineering stage during which the plan is develped, and a production stage, where the plan is executed.

Work breakdown structures:-

→ this is the architecture of the project plan.

→ A project is said to be success, if we maintain good work breakdown structure.

→ AWBS is simply a hierarchy of Element that decomposes the project plan into discrete work tasks and it provides:

1) A pictorial description of all significant work

2) A clear task decomposition for assignment of responsibilites.

3) A frame work for scheduling, budgeting, and expenditure tracking.

Conventional WBS issues:-

→ conventional work Break down structures commonly suffer from 3 fundamental faults.

1) Conventional WBS are prematurely structed around the product design.

2) decomposed, planned, and budgeted in Either too little or too much details.

3) CWBS are project specific, cross-project comparisions are usually difficult or impossible.

management

System requirements and design subsystem 1

Component 1.1

{Req, Des; code, Test, Doc ..}

component 1.N

{Req, Des, Code, Test, Doc, ..}

Sub system M

-do-

integration and test

{Test planning, Testing, Test reports}

other support areas

{Conf control, Quality assurance}.

3) major milestones:-

→ It is an iterative model, the major milestones are used to achieve concurence among all stakeholders on the current state of the project.

→ diff stakeholders have diff Concerns:

customers:- schedule and budget estimates, feasibility, risk assessment, requirements understanding, progress, product line compatibility

Users:- consistency with req & usage scenarios, potential for accommodating growth, quality attributes.

Architects & Systems Engineers:- product line compatibility, requirements changes, trade-off analyses, completeness & consistency, bal among risk, quality & usability.

Developers:- sufficiency of req detail & usage scenario descriptions, frame work for component selection or development, resolution of development risk, product line compatibility sufficiency of the development Environment.

maintainers:- sufficiency of product & documentation artifacts, understandability, interoperability with existing systems, sufficiency of maintenance Environment.

others:- regulatory agencies venture capital investors, subcontractors, associate contractors.

and sales & marketing team.

* lifecycle objective milestone:- these milestone occur at the end of the inception phase.

→ the goal is to present to all stakeholders a recommendation on how to procced with developmend, including a plan, estimated cost & schedule, and expected benefits & cost savings.

* life cycle Architecture milestone:- these milestone occures at the end of the Elaboration phase.

→ primary goal is to demonstrate an executable architecture.

* initial operational capability milestone:-

→ these milestones occur late in the construction phase.

→ the goals are to assess the readiness of the s/w to begin the transition into customer/user sites & to authorize the start of acceptance testing.

* product release milestone:-

→ occures at the end of the transition phase.

→ the goal is to assess the completion of the s/w & its transition to the support org, if any.

4) Periodic Status Assessments :-

→ these are management reviews conducted at regular intervals (monthly, quarterly) to address progress & quality of project & maintain open communication among all stake holders.

→ the main obj of this assessment is to synchronize all stake holders expectations and also serve as project snapshots.

Also provide,

1) A mechanism for openly addressing, com^{n}icating & resolving managment issues, tech issues, & project risks.

2) A mechanism for broadcast process, progress, quality trends, practices & experience information to and from all stake holders in an open forum.

3) objective data derived directly from on-going activities & Evolving product configurations.

5) planning guidlines:-

→ s/w projects span a broad range of application domains.

→ It is valuable but risky to make specific planning suggestions independent of project context.

→ planning provides a skeleton of the project from which the management people can decide the starting ~~plann~~ point of the project.

→ In order to proper plan it is necessary to capture the planning guidelines from most expertise & experience people.

→ project-independent planning advice is also risky.

*Two simple guidelines when a project is initiated or assessed:-

1) A default allocation of costs among the first-level WBS elements:

first-level WBS Elements	Default Budget
management	10%
Environments	10%.

Requirements	10%
Design	15%
implementation	25%
assessment	25%
Deployment	5%
Total	100%

→ the above table provides default allocation for budgeted costs of each first-level WBS element.

→ Sometimes these values may vary across projects but this allocation provides a good benchmark for assessing the plan.

→ It is cost allocation table not the effort allocation.

2) Allocation of Effort & Schedule across the life cycle phases

Domain	Inception	Elaboration	constr	Transition
Effort	5%	20%	65%	10%
Schedule	10%	30%	50%	10%

→ the above table provides guidelines for allocating effort & schedule across the life-cycle phases.

→ these Values can also Vary widely, depending on the specific constraints of an application they provide an average Expectation across a spectrum of appliaation domains.

6) Cost & Schedule Estimating process:-

→ project plans need to be derived from two perspectives:-

1) forward-looking, top-down approach:-

→ It starts with an understanding requirements & constraints derives a macro-level budget & schedule, then decomposes these Elements into lower level budgets & intermediate milestones.

→ from this perspective the following planning sequences would occur:

a) the s/w project manager develops a characterization of the overall size, process, Environment, People, & quality required for the project.

b) A macro-level Estimate of the total effort & schedule is developed using a s/w cost Estimation model.

c) the s/w project manager partitions the estimate for the effort into a top-level WBS using guidelines & also partitions the Schedule into major milestone dates & partition the Effort into a staffing profile using guidelines.

d) sub project managers are given the responsibility for decomposing Each of the WBS Elements into lower levels using their top-level allocation, staffing profile, & major milestone dates as constraints.

2) Backward-looking, bottom-up approach :-

→ we start with the End in mind, analyze the micro-level budgets & schedules, then sum all these Elements into higher level budgets & intermediate milestones.

→ this approach tend to define the WBS from the lowest levels upward. from this perspective, the following planning sequence would occur:-

a) the lowest level WBS Elements are Elaborated into detailed tasks.

b) Estimates are combined & integrated into higher level budgets and milestones.

c) comparisions are made with the top-down budgets & schedule milestones.

→ these two planning approaches should be used together, in balance, throughout the life cycle of the project.

→ During the Engineering stage, the top-down perspective will dominate becoz there is usually not enough depth of understanding nor stability in the detailed task sequences to perform credible bottom-up planning.

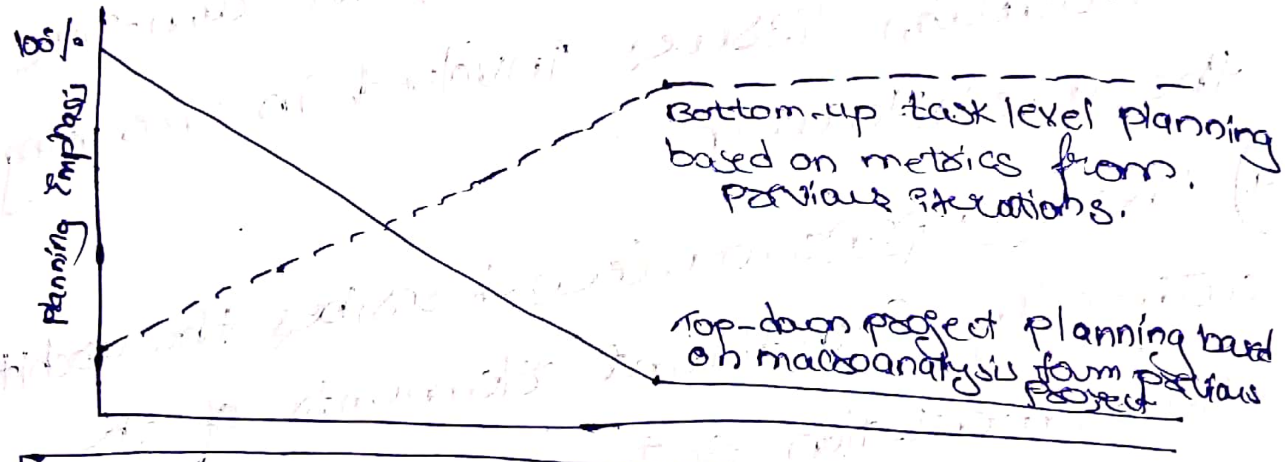
→ During the production stage, there should be enough experience & planning fidelity that bottom-up planning perspective will dominate.

→ By then, the top-down approach should be well tuned to the project specific parameters, so it should be used more as a global assessment technique.

7) the iteration planning process:-

→ so far, this discussion has deal to only with the application - independent aspects of the budgeting & scheduling.

→ Another dimension of planning is concerned with defining the actual sequence of intermediate results.



Engineering stage		Production stage.	
inception	Elaboration.	construction	Transition

→ the design set includes all UML design models describing the solution space.

→ the design, process, and use case models provide for visualization of the logical & behavioral aspects of the design.

→ the component model provides for visualization of the implementation set.

→ the deployment model provides for vis. of the deploy ment set.

1) the usecase view describes how the system's critical use cases are realized by elements of the design model.

2) the design view describes the architecturally significant elements of the design model.

3) the process view addresses the run-time collaboration issues involved in executing the architecture.

4) the component view describes the architeculturally significant elements of the implementation set.

→ Architecture descriptions take on diff forms & styles in diff org & domains.

→ An architecture baseline is defined as a balanced subset of information across all sets.

→ Generally architecture base line includes:

1) Requirements

2) Design

3) Implementation

4) Deployment.

8) Pragmatic Planning:-

→ Even though good planning is more dynamic in an iterative process, doing it accurately is far Easier.

→ while Executing iteration N of any phase, the s/w project manager must be monitoring & controlling against a plan that was initiated in iteration N-1 & must be planning iteration N+1.

→ the art of good project management is to make trade-offs in the current iteration plan & the next iteration plan based on the obj. results in the current iteration & previous iterations.

→ A side from bad architectures & misunderstood requirements, inadequate planning is one of the most common reason for project failures.

→ while a planning document is not very useful as an End item, the act of planning is extremely important to project Success.

- → plans are not just for managers.
- → the more open & visible the planning process & results, the more ownership there is among the team members who need to Execute it.
- → Bad, closely held plans cause attrition.
- → good, open plans can Shape cultures & Encourage team work.