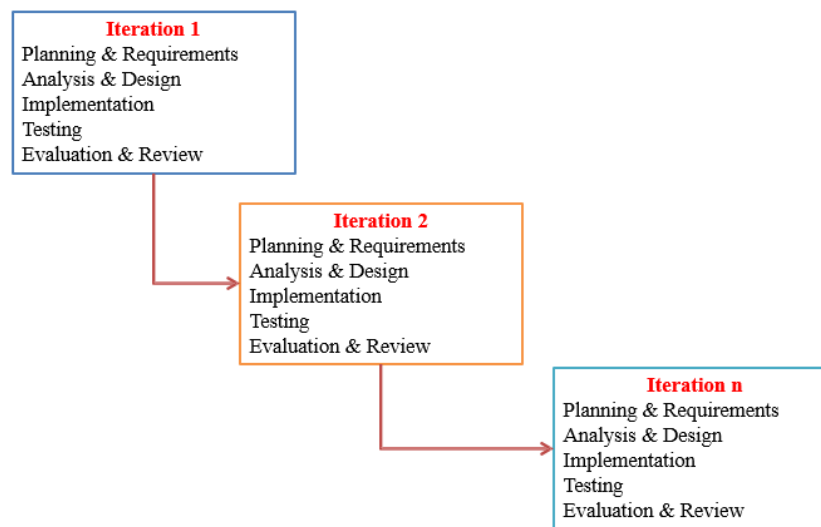


Iterative Development:

- The iterative process model is a cyclical process in which you make and test incremental adjustments. The iterative model is popular in technology, engineering, software development, design, qualitative research, project management (especially in Agile and Scrum), and more.
- At the most basic level, the process relies on a continual cycle of planning, analysis, implementation, testing, and evaluation. The iterative process starts with initial planning and overall requirements. Then the first, prioritized portion of the project becomes the initial cycle of development. That segment is refined by trial and error. Once finished, it forms the basis for the next chunk of the project. Each cycle improves on the overall product or project. The pace of this process is related to how effectively you work through the cycle. The iterative development process is a five steps process.



1. Planning and Requirements:

In this stage, map out the initial requirements, gather the related documents, and create a plan and timeline for the first iterative cycle.

2. Analysis and Design:

Create a working architecture, schematic, or algorithm that satisfies your requirements.

3. Implementation:

Develop the functionality and design required to meet the specifications.

4. Testing:

Identify and locate what's not working or performing to expectations. Stakeholders, users, and product testers weigh in with their experience.

5. Evaluation and Review:

Compare this iteration with the requirements and expectations.

- After we complete these steps, it's time to tackle the next cycle. In the iterative process, the product goes back to step one to build on what's working. Identify what you learned from the previous iteration.
- This iterative development, sometimes called circular or evolutionary development, is based on refining the first version through subsequent cycles, especially as you gather and include requirements.
- It allows you to remain flexible as you identify new needs or unexpected business issues.

Unified Process (UP)

- A **software development process** describes an approach to building, deploying, and possibly maintaining software.
- The **Unified Process** is an iterative and incremental software development process for building object-oriented systems.
- Emerged as popular software development process.
- **It reduces unexpected development cost and prevent wastage of resources.**
- Also known as **Rational Unified Process**.
- It provides a disciplined approach to assign tasks and responsibilities within a development organization.
- **Goal: To ensure the production of high-quality software that meets the needs of its end-users, within a predictable schedule and budget.**
- Unified Process (UP) is a relatively popular iterative process for projects using OOA/D, UP is common and promotes widely recognized best practices; it's useful for industry professionals to know it.
- In Unified Process, Process is a set of activities intended to reach a goal.
- In Unified Process, The inputs to the software process are the needs of the business and the output will be the software product.
- The Unified Process is one such lifecycle approach well-suited to the UML.
- Unified Process provides a disciplined approach on how to assign tasks and responsibilities within a software development organization.

Three reason behind using UP are:

1. The UP is an iterative process.
2. UP practices provide an example structure for how to do and thus how to explain OOA/D.
3. The UP is flexible, and can be applied in a lightweight and agile approach that includes practices from other agile methods (such as XP or Scrum).

UP project organizes the work and iterations across four major phases:

1. **Inception:**

- Development of a good idea into a vision of an end product.
- During the inception phase, we develop business model for the project.
- **Customer requirement identification**
- **Communication (with customers/team members) and planning.**
- **Identification of project scope**
- **Project Plan, project goal and risk identifications are made and identified.**
- The feasibility study is performed as well as the overall scope and size of the project is determined during the inception phase.
- The actors of the system and their interaction with the system are analyzed at a high level. At the end of the Inception stage, the following objectives are to be achieved:
 - Concurrence on the scope of the project and the estimates
 - Understanding of the requirements

2. **Elaboration:**

- **Elaboration means describing something in detail.**
- **In this phase, we go into more detail of inception phase.**
- **Redefine/cancel the project if needed.**
- Refined vision, iterative implementation of the core architecture, resolution of high risks, identification of most requirements and scope, more realistic estimates.
- In the elaboration phase, a baseline architecture is established, the project plan is developed and risk assessment is also performed.

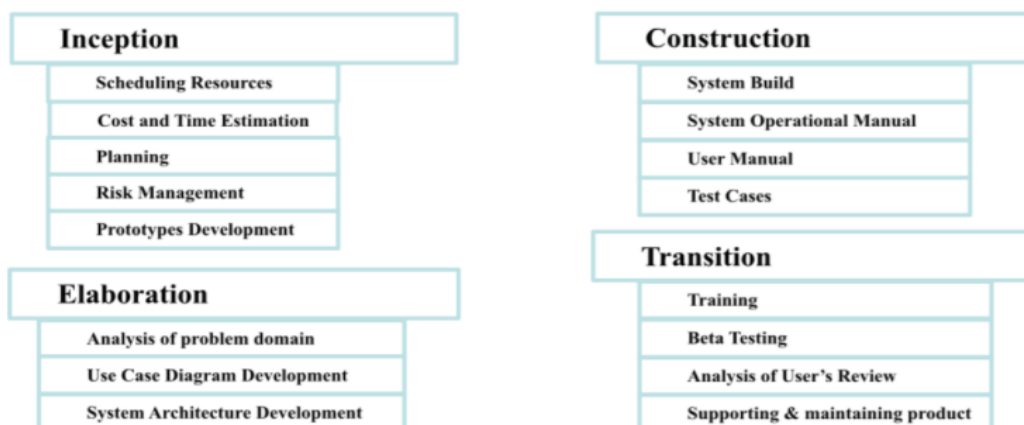
- The major types of risks are:
 - Requirements Risks
 - Technological Risks
 - Skills Risks
 - Political Risks
- At the end of the elaboration:
 - The use case model should be complete.
 - Nonfunctional Requirements should be elaborated.
 - Software Architecture should be described.
 - Revised risk list should be present.
 - A preliminary user manual (optional).

3. Construction:

- **Here we develop and complete the project based on the data we get from previous stages.**
- **Coding of the project is done here.**
- **All kind of testing are done here.[except beta testing]**
- Iterative implementation of the remaining lower risk and easier elements, and preparation for deployment.
- All the components are developed and the components are integrated during the construction phase.
- All the features are completely tested during this stage. Resources are managed and operations are controlled to optimize cost, schedule and quality.
- The construction phase is incremental and iterative. Refactoring is done after every iteration.
- At the end of the construction:
 - The product should be stable and mature for release.
 - Actual versus planned expenditure should be acceptable.

4. Transition:

- **Here finally project is transit from development environment to production.**
- **Here we set the project on beta testing mode.**
- **Remove the bugs from project based on customer feedback.**
- The objective of this phase is to transition the software product to the user community.
- New releases, correcting defects and optimization are part of this phase.
- The objectives of the transition phase are:
 - Customer Satisfaction
 - Achieving the concurrence of the stakeholders that the deployment baselines are complete and consistent with the evaluation criteria
 - Achieving final product baseline rapidly in a cost effective manner



Schedule oriented terms in UP

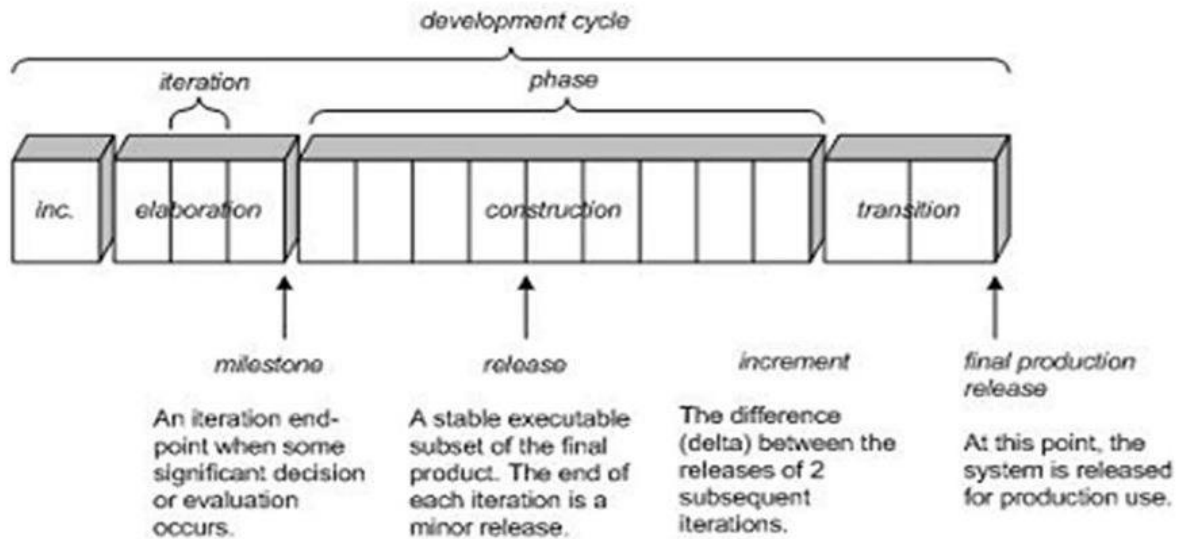
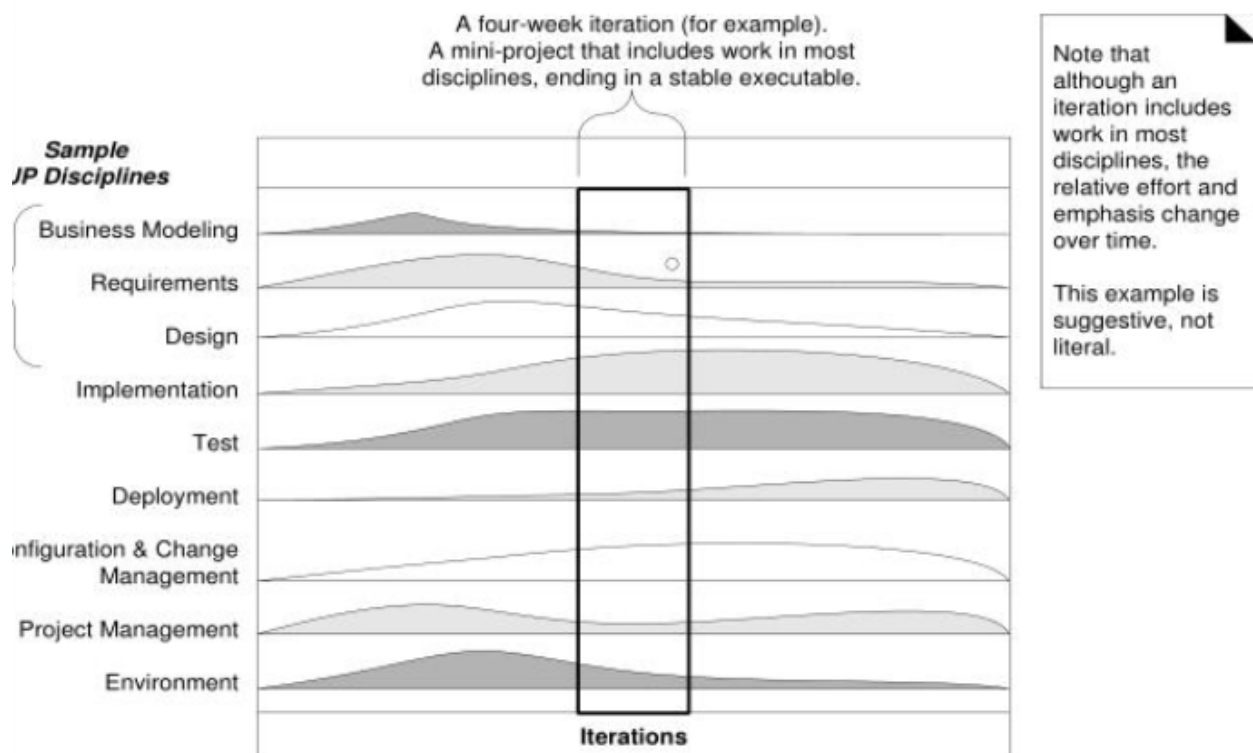


Fig: Schedule-oriented terms in the UP.

UP disciplines

The UP describes work activities, such as writing a use case, within **disciplines** a set of activities (and related artifacts) in one subject area, such as the activities within requirements analysis. In the UP, an **artifact** is the general term for any work product: code, Web graphics, database schema, text documents, diagrams, models, and so on.



- **Business Modeling:**

Purpose: understand business environment

Three major activities part of business modeling:

- Understand surroundings
- Create the system vision
- Create business models

- **Requirements:**

Objective: document business requirements

Key drivers of activities:

- discovery and understanding
- Requirements discipline and business modeling map to traditional systems analysis

Activities list:

- Gather detailed information
- Define functional and nonfunctional requirements
- Develop user interface dialogs Evaluate requirements with users

- **Design:**

Objective: design system based on requirements

Six major activities in the design discipline:

- Design support services architecture and deployment environment
- Design the software architecture
- Design use case realizations
- Design the database
- Design the system and user interfaces
- Design the system security and controls

- **Implementation:**

Objective: build or acquire needed system components

Implementation activities:

Build software components

Acquire software components

Integrate software components

- **Testing:**

Testing is critical discipline

Testing activities:

- Define and conduct unit testing
- Define and conduct integration testing
- Define and conduct usability testing
- Define and conduct user acceptance testing

In UP, acceptance testing occurs throughout the building phase

- **Deployment:**

Goal: conduct activities to make system operational

Deployment activities:

- Acquire hardware and system software
- Package and install components
- Train users
- Convert and initialize data

- **Project Management:**

Most important support discipline

Project management activities:

- Finalize the system and project scope
- Develop the project and iteration schedule
- Identify project risks and confirm feasibility
- Monitor and control the project's plan
- Monitor and control communications
- Monitor and control risks and outstanding issues

- **Configuration and Change Management:**

Configuration and change discipline pertains to:

- Requirements
- Design
- Source code
- Executables

The two activities in this discipline:

- Develop change control procedures
- Manage models and software components

- **Environment:**

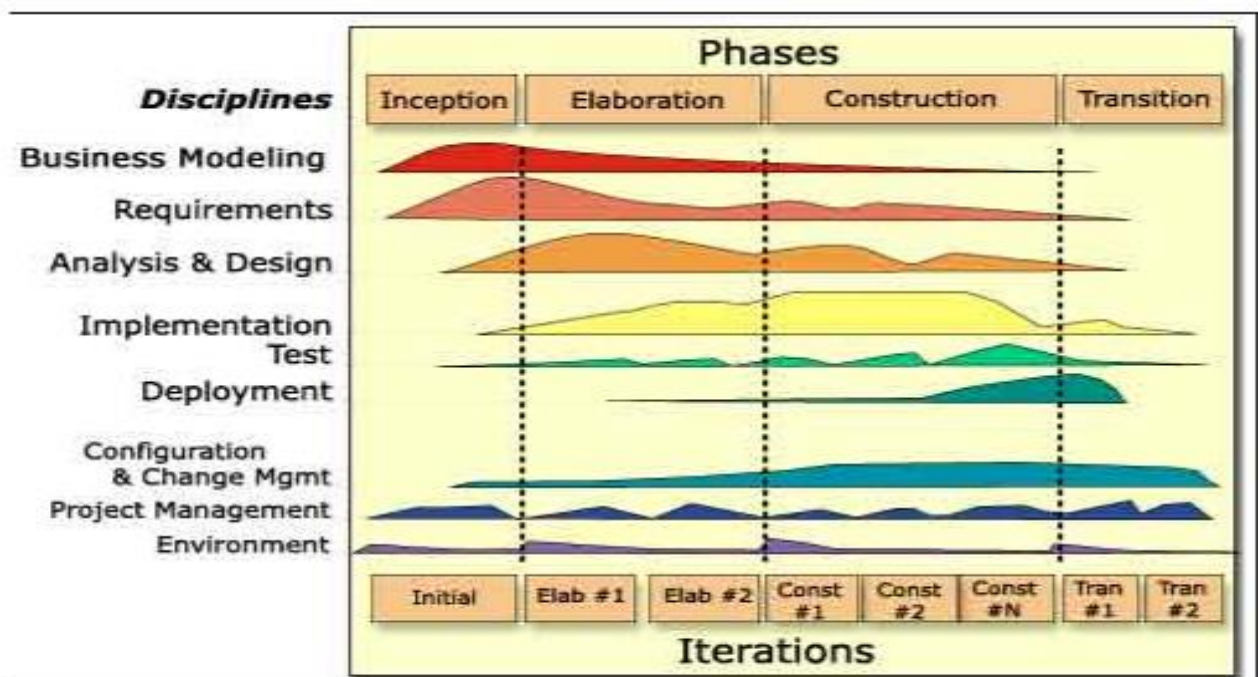
Development environment includes:

- Available facilities
- Design of the workspace
- Forums for team communication and interaction

Environment discipline activities:

- Select and configure the development tools
- Tailor the UP development process
- Provide technical support services

Relationship between UP disciplines/workflows and Phases



- In **elaboration** the iterations have a relatively high level of requirements and design work, with some implementation.
- During **construction**, the emphasis is heavier on implementation and lighter on requirements analysis.