UNIT – 4 7 Hours

PROCESS OVERVIEW, SYSTEM CONCEPTION, DOMAIN ANALYSIS

Syllabus :

- **Process Overview: Development stages; Development life cycle. System Conception:**
- **Devising a system concept; Elaborating a concept;**
- **Preparing a problem statement. Domain Analysis: Overview of analysis;**
- **Domain class model; Domain state model; Domain interaction model;**
- **Iterating the analysis.**

Process overview

- A ***software development process*** provides a basis for the organized production of software, using a collection of predefined techniques and notations.

Development Stages

- System Conception
  - Conceive an application and formulate tentative requirements
- Analysis
  - Deeply understand the requirements by constructing models
- System design
  - Devise the architecture
- Class design
  - Determine the algorithms for realizing the operations
- Implementation
  - Translate the design into programming code and database structures
- Testing
  - Ensure that the application is suitable for actual use and actually satisfies requirements
- Training
  - Help users master the new application
- Deployment
  - Place the application in the field and gracefully cut over from legacy application
- Maintenance

Preserve the long term viability of the application

Analysis

To specify *what* must be done.

- **Domain analysis** focuses on real-world things whose semantics the application captures.

Application analysis addresses the computer aspects of the application that are visible to users

System Design

- Devise a high-level strategy — the architecture — for solving the application problem.
- The choice of architecture is based on the requirements as well as past experience.

Class Design

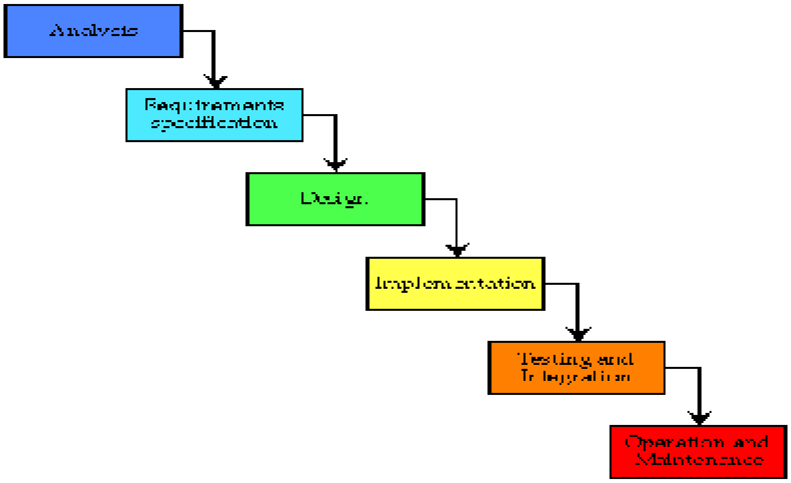
- To emphasis from application concepts toward computer concepts.
- To choose algorithms to implement major system functions.

Development Life Cycle

- Waterfall Development
- Iterative Development

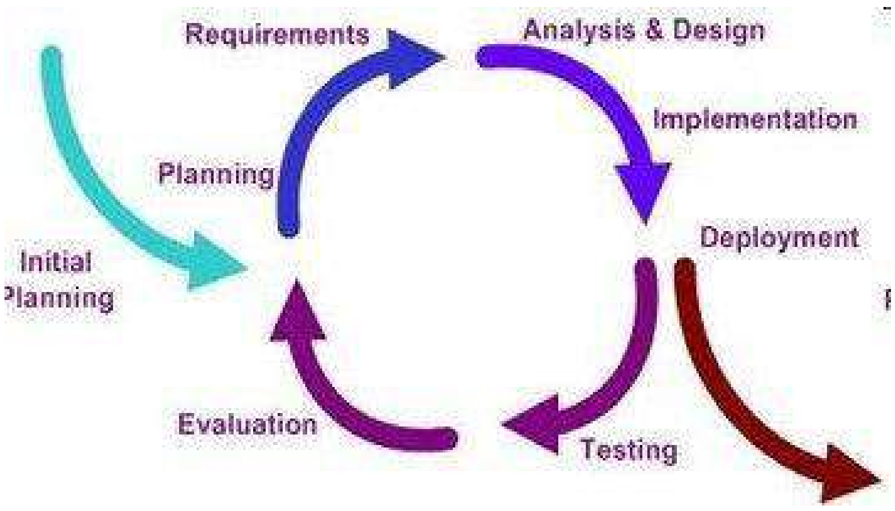
Waterfall Development

- The stages in a **rigid linear sequence** with no backtracking.
- Suitable for well-understood applications with predictable outputs from analysis and design.

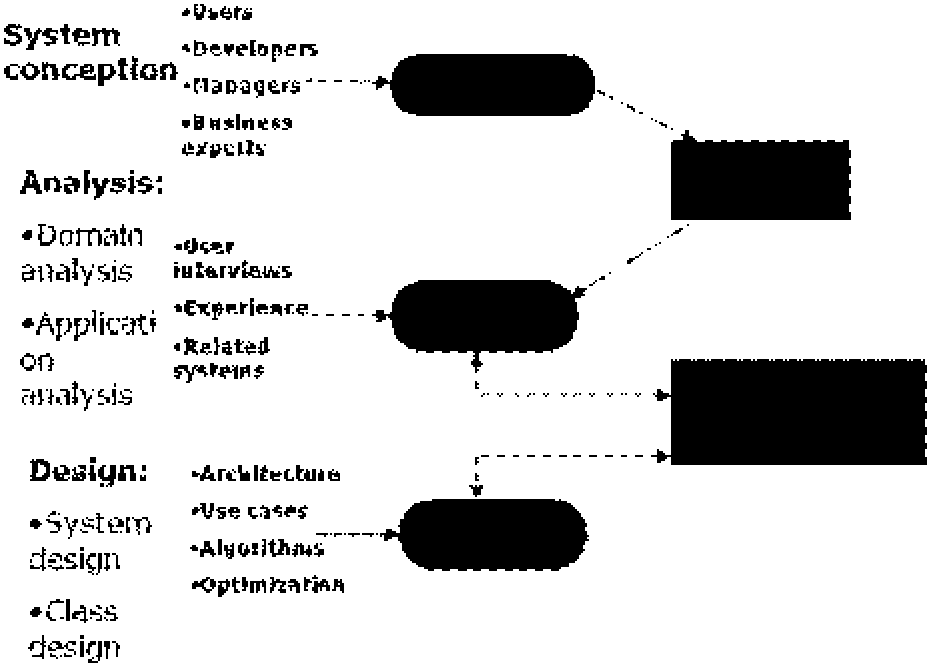


Iterative Development

- First develop the **nucleus of a system**, then grow the scope of the system…
- There are **multiple iterations** as the system evolves to the final deliverable.
- Each iteration includes a full complement of stages:
  - analysis, design, implementation, and testing



Summary of development process for the organized production of software



System Conception

- System conception deals with the genesis of an application

Devising a System Concept

- New functionality
- Streamlining
- Simplification automate manual process
- Integration
- Analogies
- Globalization

Elaborating a Concept

Good system concept must answer the following questions

- Who is the application for?
    - Stakeholders of the system
- What problems will it solve?
    - Features
- Where will it be used?
    - Compliment the existing base, locally, distributed, customer base
- When is it needed?
    - Feasible time, required time
- Why is it needed?
    - Business case
- How will it work?
    - Brainstorm the feasibility of the problem

The ATM Case Study

Develop software so that customers can access a bank's computers and carry out their own financial transactions without the mediation of a bank employee.

The ATM Case Study

- Who is the application for?
    - We are vendor building the software
- What problems will it solve?
    - Serve both bank and user
- Where will it be used?
    - Locations throughout the world
- When is it needed?
    - Revenue , investment
- Why is it needed?
    - Economic incentive. We have to demonstrate the techniques in the book
- How will it work
    - N-tier architecture, 3-tier architecture

Preparing a problem statement

Design the software to support a computerized banking network including both human cashiers and automatic teller machines (ATMs)to be shared by a consortium of banks. Each bank provides its own computer to maintain own accounts and process transactions against them. Cashier stations are owned by individual banks and communicate directly with their own bank’s computers. Human cashiers enter account and transaction data ......

The ATM Case Study

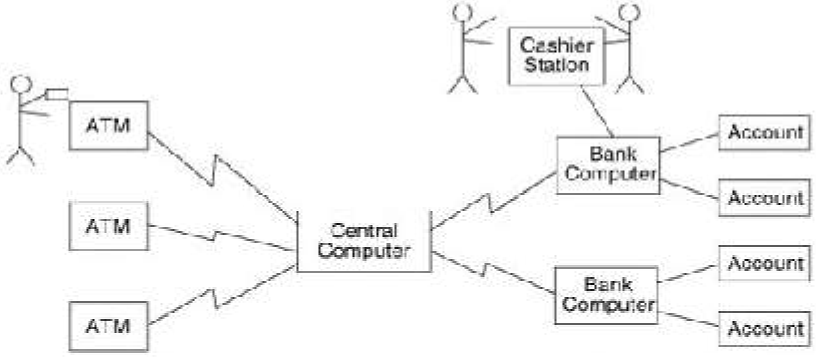


Figure 11.3 ATM network. The ATM case study threads throughout the remainder of this book