

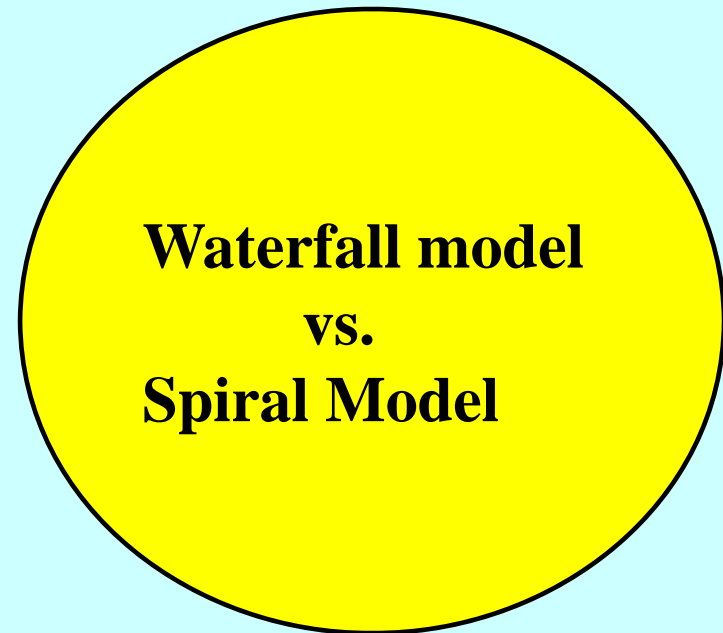
Requirements Analysis (Chap-11) and Requirements Specifications (Chap-12)

Requirements Analysis

Chap-11

Software Development Activities

- **Requirements Analysis**
- **Design**
- **Implementation**
- **Testing**
- **Acceptance Test**
- **Production**
- **Modification and Maintenance**



Requirements Analysis

- **create precise description of needs of customer**
- **must consider both normal situations and errors**
- **Two sources of errors:**
 - **users interacting with the program**
 - **hardware and software malfunctions**

Use Case Scenarios

- **Scenario: step by step walk through of an interaction with the system, consisting of use/response pairs**
- **Work out Scenarios for both normal case behavior and behavior in presence of errors**

Functional and Non-Functional Requirements

- **Availability, Reliability**
- **Performance requirements**
 - **time and space efficiency**
 - **throughput and response time**
- **Modifiability and Reusability**

Goals of Requirements Analysis

1. Identify the functional requirements

- **how a correctly functioning program responds to both correct and incorrect user interactions**
- **how the program responds to both hardware and software errors**

Goals of Requirements Analysis (continued)

2. Identify performance requirements

- how fast certain actions must be, and any constraints on the amount of storage that can be used**

3. Identify potential Modifications

4. Pin down the delivery schedule

Output of Requirements Analysis phase: Requirements Document

Requirements Document contains:

- **the requirements specification (Chap-12)**
- **desc. of performance requirements, potential modifications, scheduling constraints...**
- **discussion of alternatives that were considered and rationale for decisions**

Requirements Document

Requirements Document is input to:

- **Software Design Phase**
- **produce Acceptance Tests**
- **basis for User's Manual**

Example: Stock Tracker

- **Keeps track of user investments**
- **Stores information about all stocks owned by a user**
 - **amount owned**
 - **date and price when purchased**
 - **other info**
- **Provides info on current price, total value of all stocks owned**

Consider Normal Scenarios

- **what style of interaction:**
 - **user waits for program to demand info (interactive) or**
 - **user gives it as arguments when prog. called**
- **how price info obtained:**
 - **user enters (error prone) or obtained from web server (more convenient)?**
- **how often is price info refreshed: from web server (remote communication or user request)**

Example: Normal case Scenario

- **user start by invoking the prog. (interactive or user supply args)**
- **accept user commands**
 - **identify a portfolio for further examination**
 - **browse info in identified portfolio**
 - **create or delete a portfolio**
 - **add or remove a position from a portfolio**
- **terminate session**

User errors

Include user commands that are wrong:

- **user might type the wrong ticker name**
- **user enters wrong number of shares**
- **delete a portfolio that contains investments**

Errors can be handled:

- **check whether args are correct etc.**

System malfunctions

- **How to prevent loss of information due to computer crash**
 - **each modification is recorded on disk immediately?**
 - **save on user request?**
 - **write info to a second disk?**

Performance and Potential changes

- **Performance**
 - **Remote communication?**
 - **write to second disk?**
- **Potential Changes**
 - **extend to handle mutual funds**
 - **undo facility**
 - **additional web servers (high availability)**

Requirements Specifications

(Chap-12)

Model state of entire program

- **Define program state by means of a *data model***
- ***Data model* is then used in Requirements specification**
- **Data model contains**
 - **Definitions of domain, subsets and relations**
 - **a graph that shows how sets and relations interrelate and defines constraints on them**
 -

Data Model

- ***Data model* contains:**
 - **Definitions of domain, subsets and relations**
 - **a graph that shows how sets and relations interrelate and defines constraints on them**
 - **definitions of derived relations and additional constraints**

Data Model

- **the graph defines the kind of data being manipulated and how they are related to one another**
- **graph and textual description together defines constraints on what program does.**
- **Nodes are sets, edges are relationships**

Data Model

- **The data model describes a state of the system which may change over time**
- **The model expresses this mutability in two ways:**
 - **items may be added or removed from sets**
 - **relationships between sets may change**

Domains and Subsets

- **Sets with no supersets : domain**
- **each domain is disjoint from one another**
- **Three constraints are used to define subsets:**
 - **fixed: membership is fixed for all time**
 - **static: an elem never switches between belonging to subset and not belonging to subset**
 - **how many elements a subset contains**

Relations

- **Relations are used to indicate how items of a set are related to items in other sets**
- **Each relations has a source and a target**
- **Multiplicity:**
 - **! (exactly 1), ? (0 or 1), * (0 or more), + (1 or more)**
- **Mutability: if the mapping can change**
 - **applied at both ends of arrow**

Textual Information

- **Two parts:**
 - **first part: descriptions of sets and relations**
 - **second part: additional constraints and derived relations**

Contents of a Requirements Specification

- contains specifications for *static* (when app is not already running) and *dynamic* operations (used only after the app is already running)
- if interactive program, operations are limited to string args and no results, include *checks* clause that must hold for operation to run
- operation specifications based on data model

Two issues

- **interactive or not**
- **external data formats**

Requirements Specs Example: Stock Trader

1. The Data Model:

- **Domains and relations,**
- **Graph**
- **Constraints and derived relations**

2. The Stock Trader Specifications