CP317 Software Engineering

week 8-1 - Waterfall model and its variants

Shaun Gao, Ph.D., P.Eng.

Agenda

- Review week 7-2 topics
- Introduction
- Predictive models vs. adaptive models
- Waterfall model
- Waterfall model with feedback
- Sashimi model (waterfall with overlapping phase)
- Incremental waterfall model
- V-model
- Summary

Review

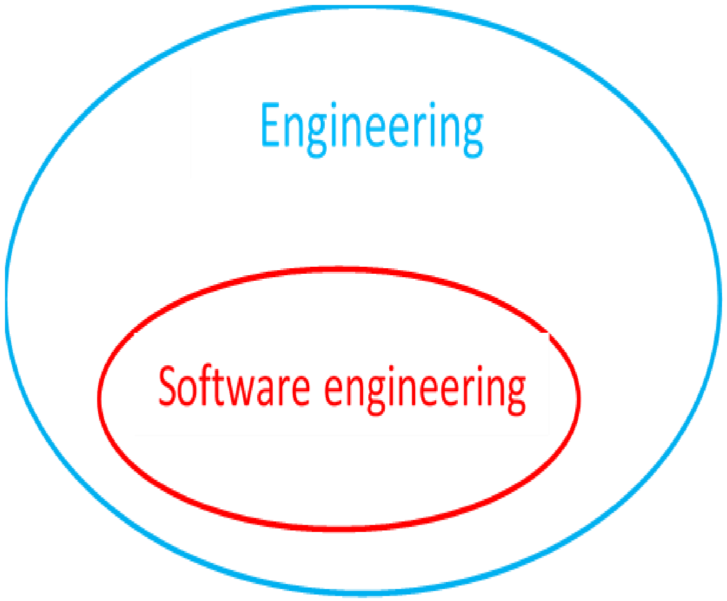
- Metrics
- Types of software metrics
    - Process metrics
    - Product metrics
    - Project metrics
- Defect analysis
- Ishikawa diagrams
- Normalization in software metrics
- Software complexity, cyclomatic complexity measure
- Function points

Introduction

- The process of software engineering
  - Project start
  - Requirement gathering and analysis
  - Architecture design (high-level design)
  - Detailed design (low-level design)
  - Implementation (code design)
  - Testing
  - Deployment
  - Maintenance
- All the comparison among SE process models
  - Assumption: same team and same manager

Introduction – cont.

- Engineering design process
  - Engineering design process is a series of steps that engineers follow to come up with a solution to a problem.
- Software engineering is a process
- Software development life cycle (SDLC)
  - SDLC is a structured step-by-step approach for developing software products.



Intangible Product

No Standard Software Process

SE Process models

- Proposed process Models:
  - Waterfall model
  - Iterative model
  - Spiral model
  - Extreme programming (XP)
  - Agile methodology
  - Rapid application development (RAD)
- Reasons why software engineering models are needed
  - Standardization
  - Optimization

Benefits of using process models

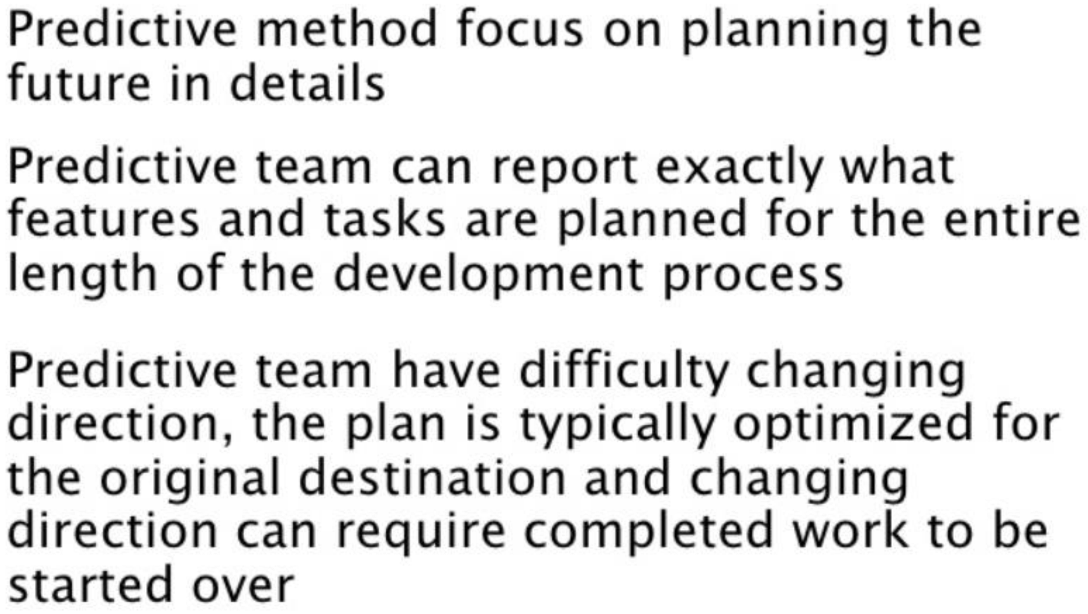
- Quality
- Mutual understanding
- Documentation
- Improvement/Optimization
- **Standardization**
- Viewing system from multiple perspective
- Discovering causes and effects using model traceability
- Discovering errors earlier and reducing system defects

Predictive models and Adaptive models

- A predictive model is a process model of software engineering through which a future outcome or behavior is predicted based on the past and current data at hand.
  - When project activities are divided into linear sequential tasks, anyone can predict the output of each task
  - Example: Waterfall model
- An adaptive model is a process model of software engineering that embodies the principle that continuous adaptation of the process to the work is the normal state of affairs.
  - Example: Agile approach

Predictive models vs. Adaptive models

- Predictive models

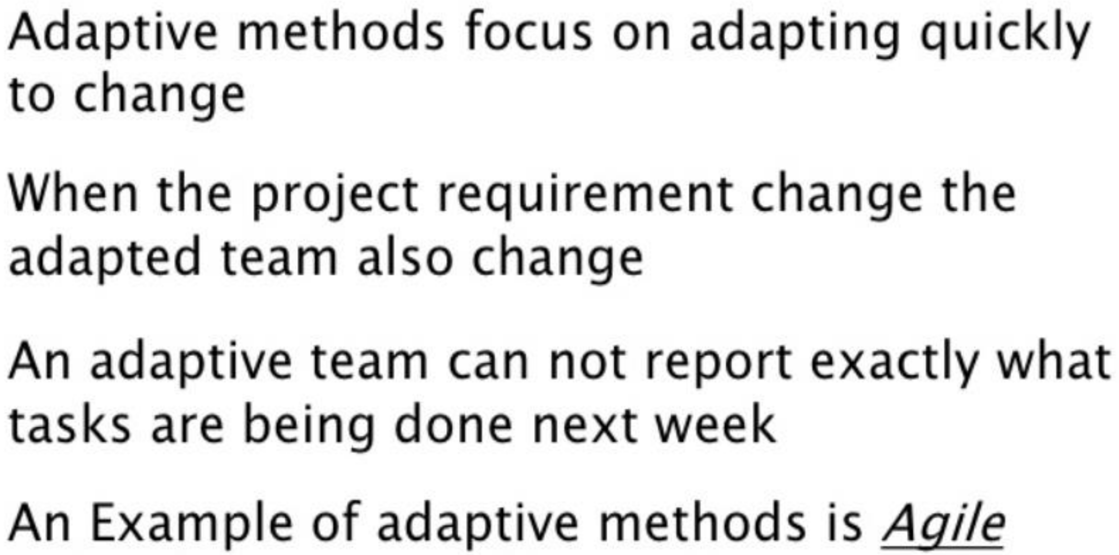


Predictive method focus on planning the future in details

Predictive team can report exactly what features and tasks are planned for the entire length of the development process

Predictive team have difficulty changing direction, the plan is typically optimized for the original destination and changing direction can require completed work to be started over

Adaptive models



Adaptive methods focus on adapting quickly to change

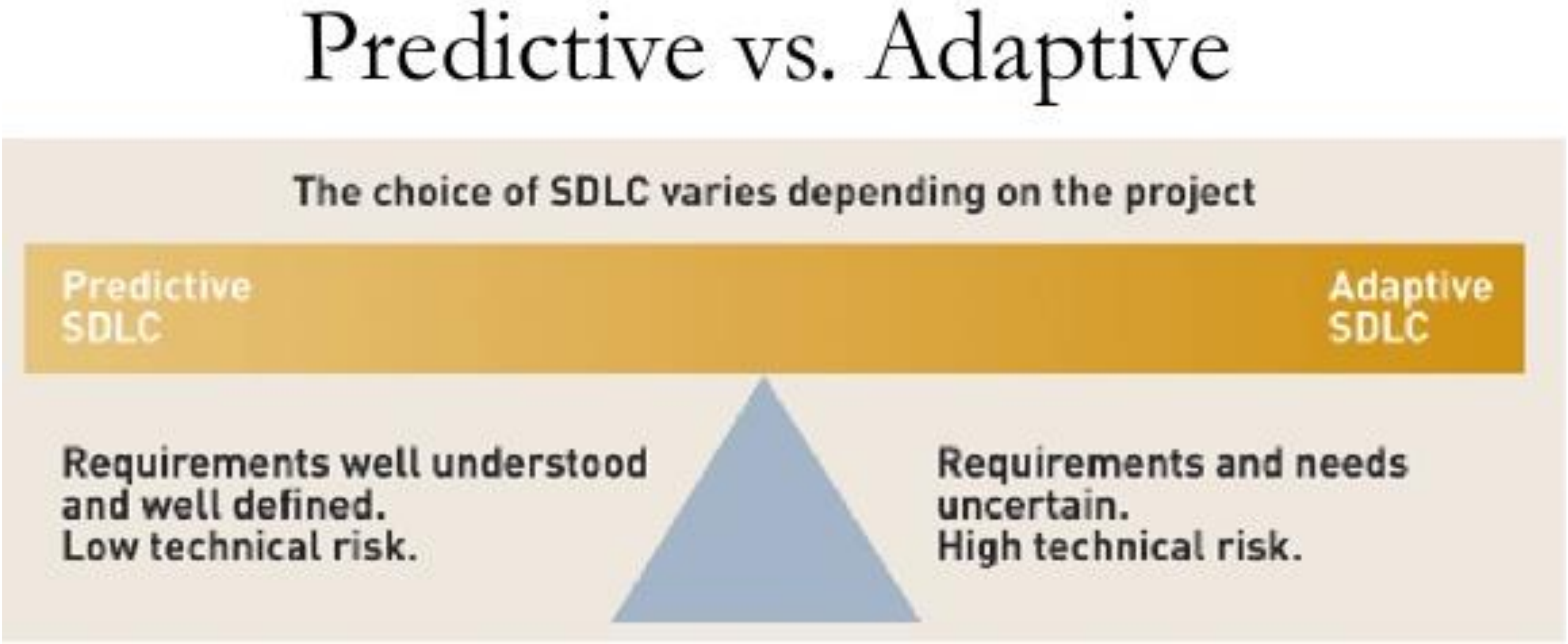
When the project requirement change the adapted team also change

An adaptive team can not report exactly what tasks are being done next week

An Example of adaptive methods is *Agile*

- **Question**? Which model is better?

Predictive models vs. Adaptive models



Predictive vs. Adaptive

The choice of SDLC varies depending on the project

Predictive
SDLC

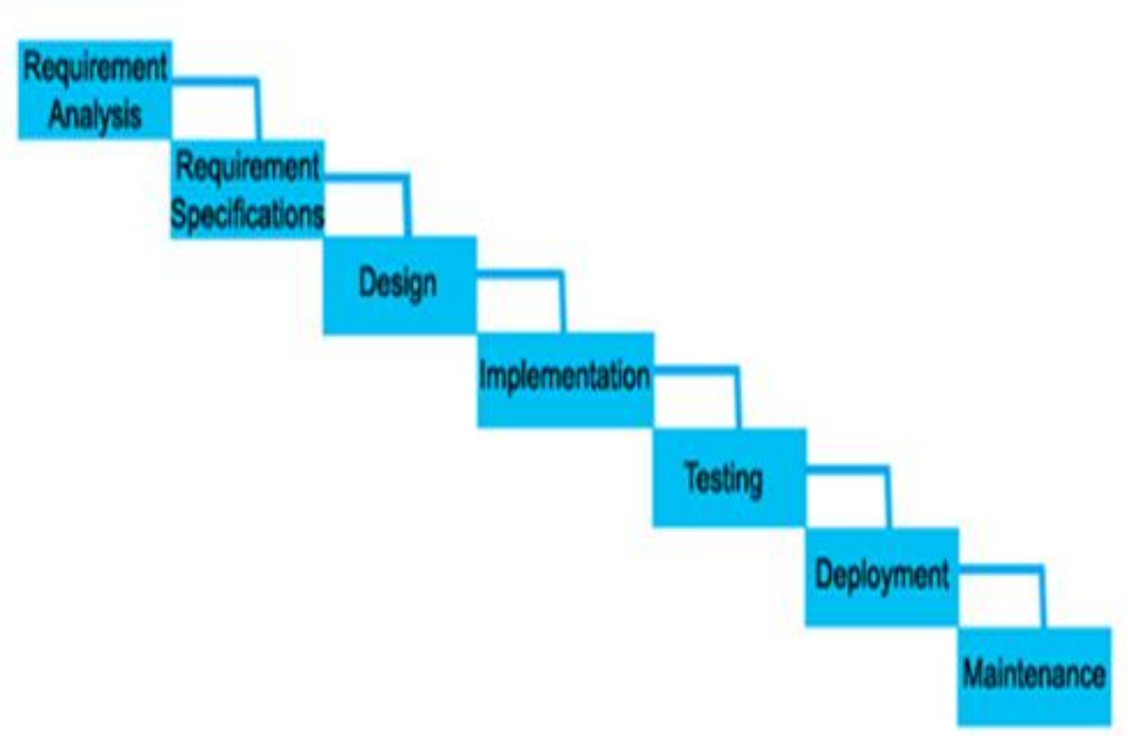
Adaptive
SDLC

Requirements well understood and well defined.
Low technical risk.

Requirements and needs uncertain.
High technical risk.

Waterfall process model

- The **waterfall model** is a breakdown of project activities into linear sequential phases, where each phase depends on the deliverables of the previous one and corresponds to a specialization of tasks.

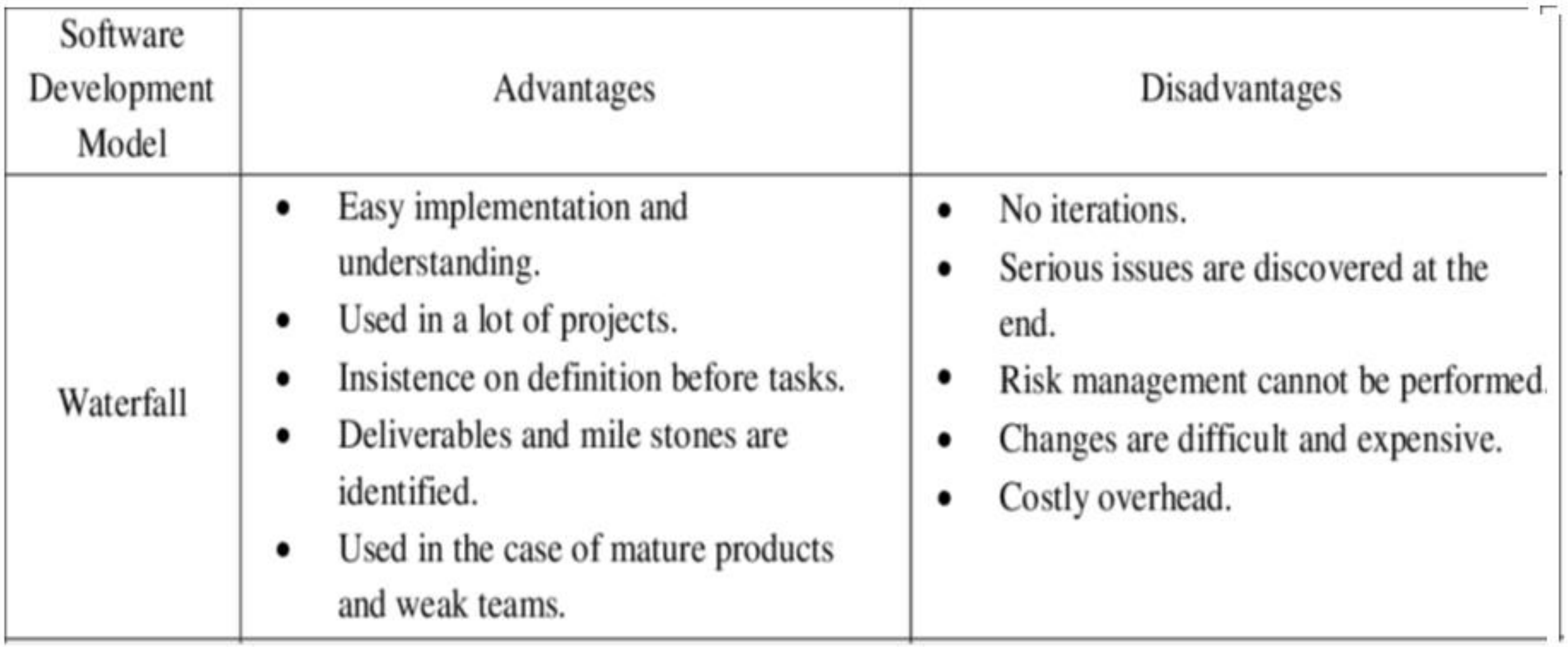


Waterfall process model –cont.

- Waterfall model can work reasonably well if
  - The requirements are precisely known in advance
  - The requirements include no unresolved high-risk items
  - The requirements will not change much during development
  - The team is knowledgeable and experienced

Waterfall process model –cont.

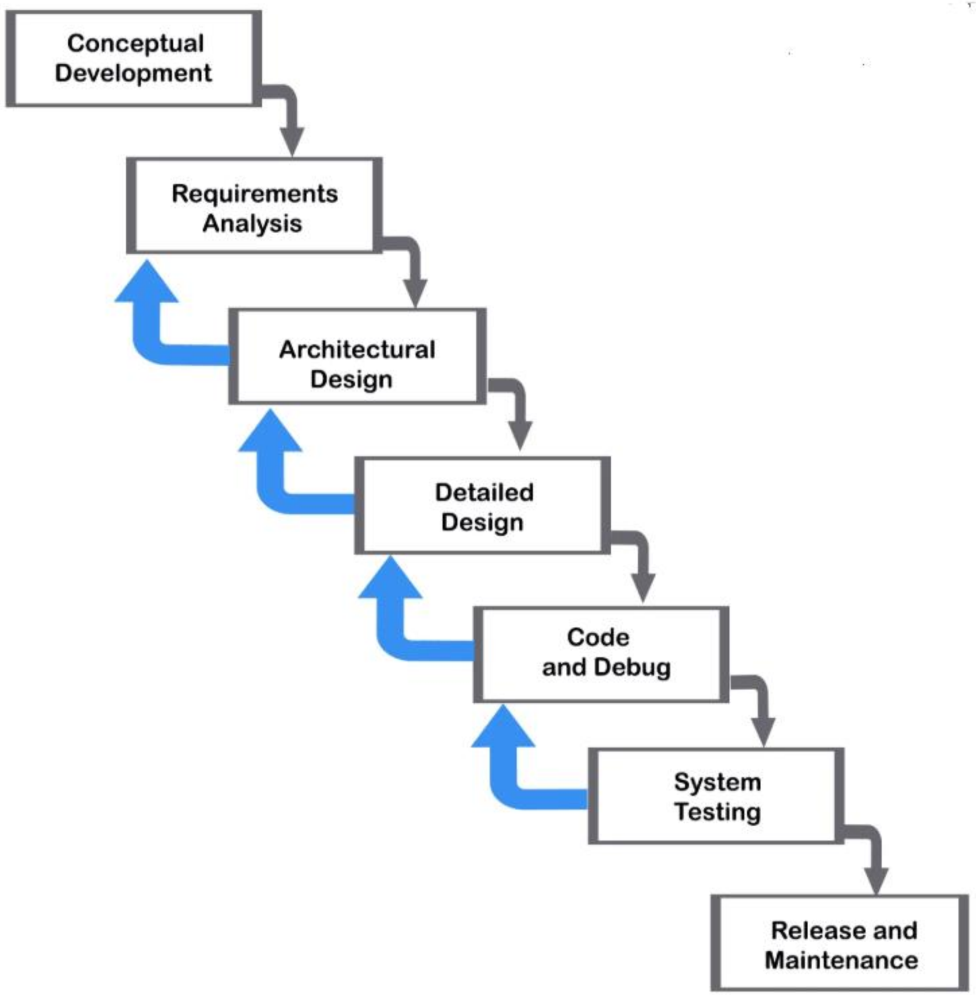
- The advantages and disadvantages of the waterfall model



Software Development Model	Advantages	Disadvantages
Waterfall	• Easy implementation and understanding. • Used in a lot of projects. • Insistence on definition before tasks. • Deliverables and mile stones are identified. • Used in the case of mature products and weak teams.	• No iterations. • Serious issues are discovered at the end. • Risk management cannot be performed. • Changes are difficult and expensive. • Costly overhead.

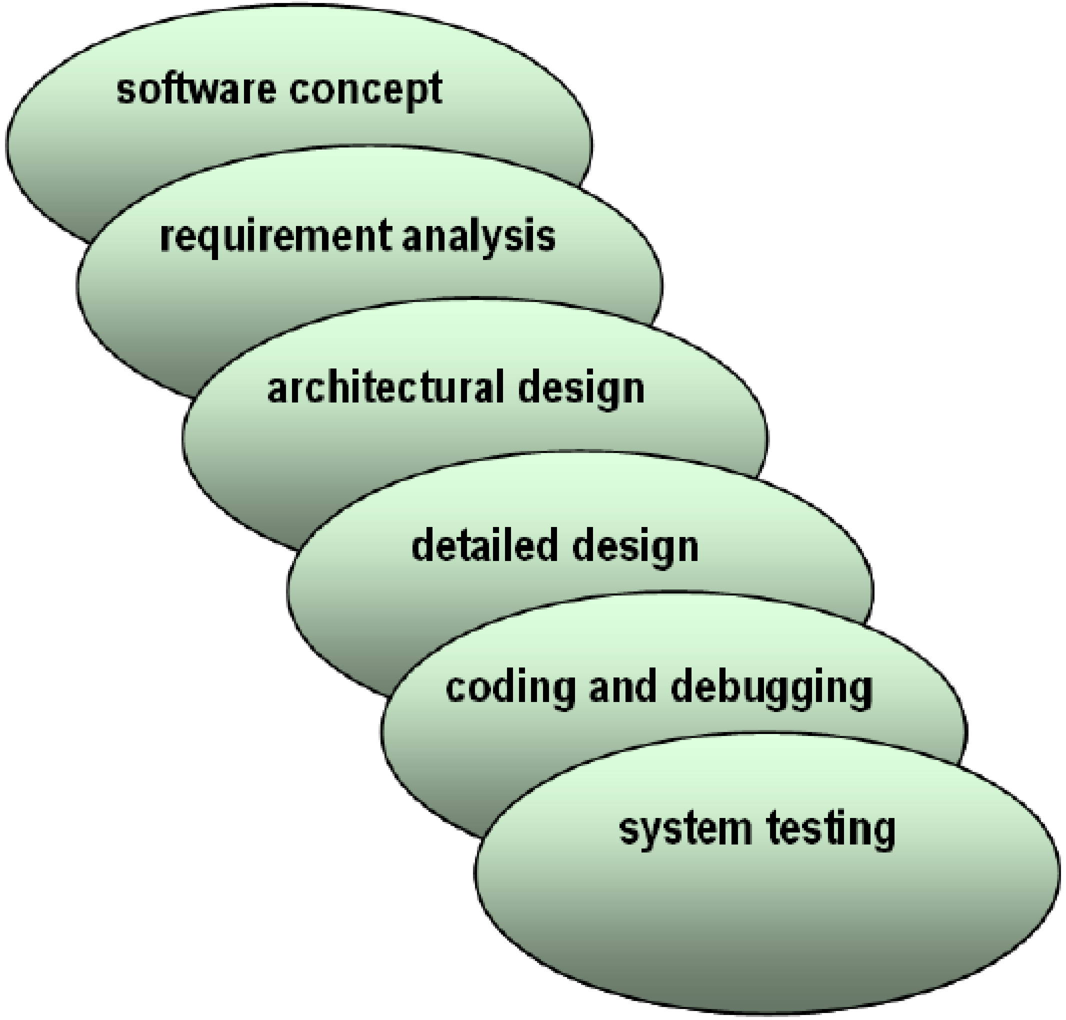
Waterfall with feedback

- Waterfall with feedback model is a variety of waterfall model, which provides abilities to back up to a previous phase when engineers or developers discover a problem in current phase.
- Waterfall with feedback model is better than pure waterfall model
  - Flexibility (because unpredicted changes are common)
  - Improve productivity



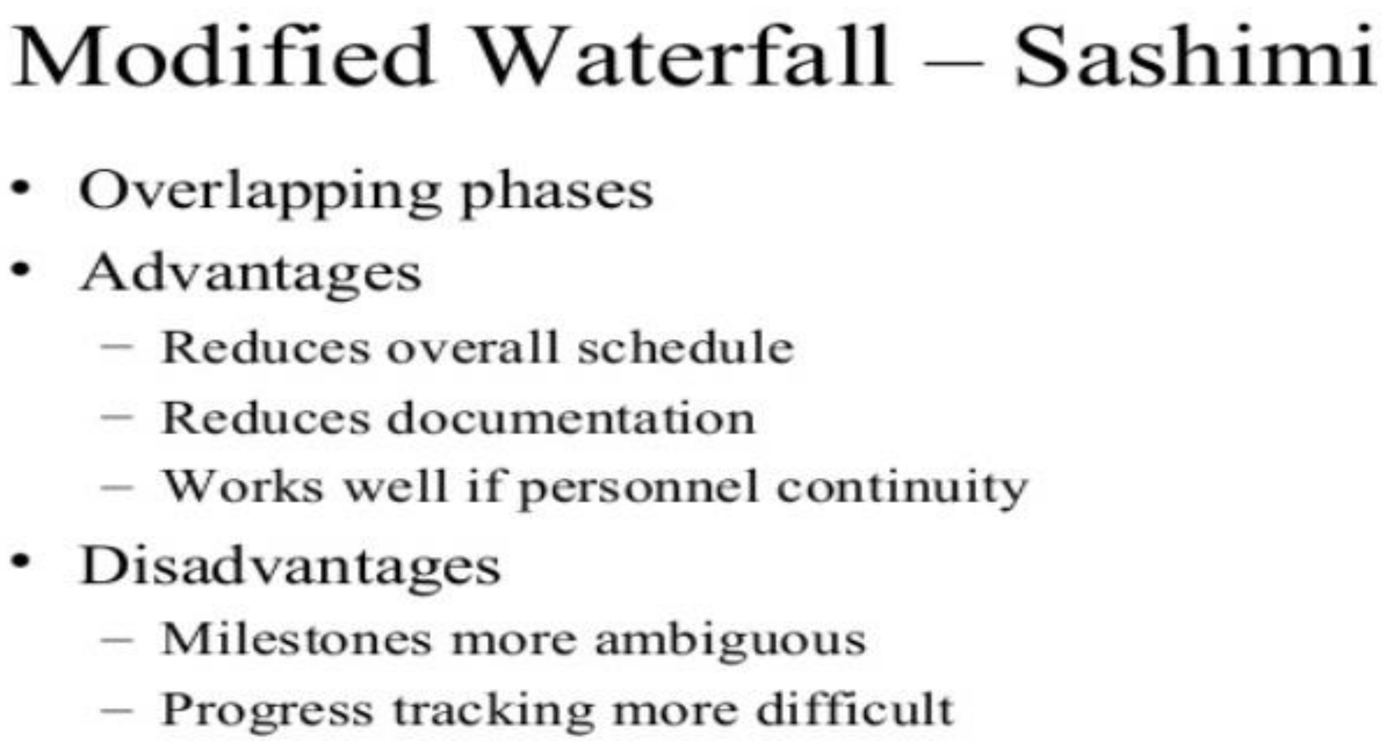
Sashimi model

- Known as waterfall with overlapping phases
  - A Sashimi model is a type of a waterfall model that has overlaps among stages.



Sashimi model – cont.

- Advantages and Disadvantages of sashimi model

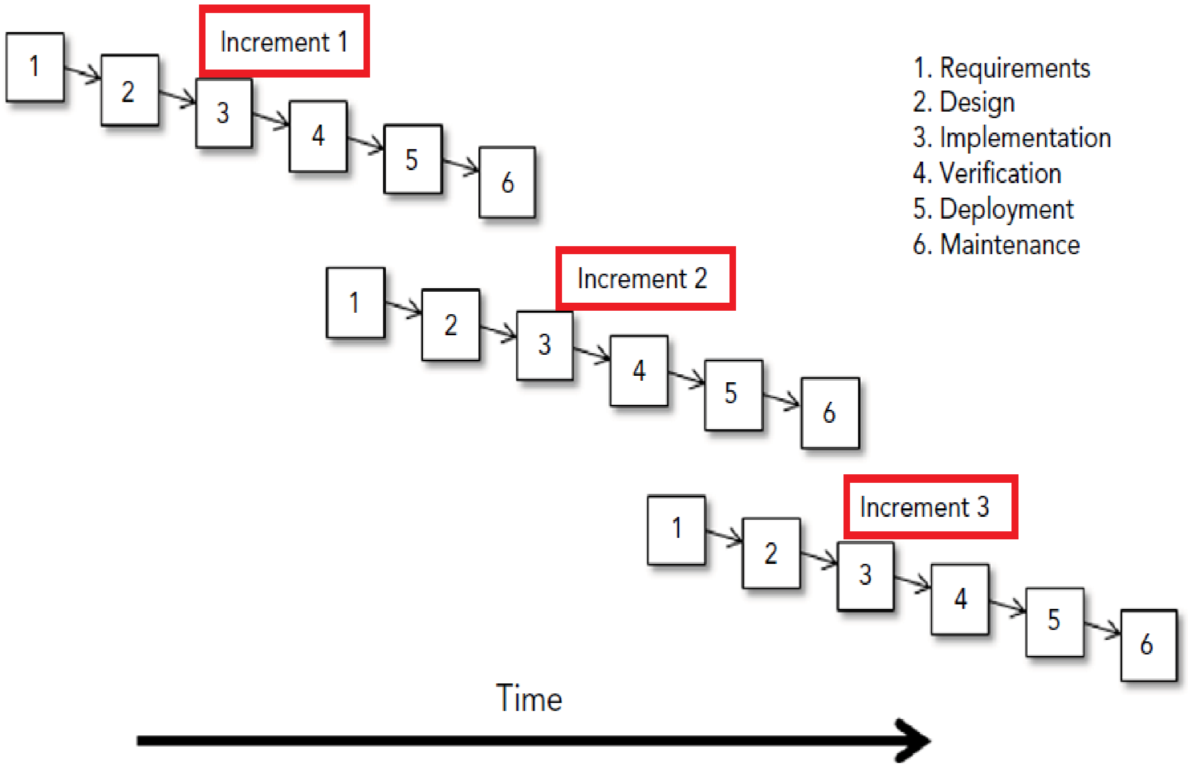


Modified Waterfall – Sashimi

- Overlapping phases
- Advantages
  - Reduces overall schedule
  - Reduces documentation
  - Works well if personnel continuity
- Disadvantages
  - Milestones more ambiguous
  - Progress tracking more difficult

Incremental waterfall model

- An incremental waterfall model is a process of software development where requirements are broken down into multiple standalone modules of the software development cycle that uses the waterfall model.



Incremental waterfall model – cont.

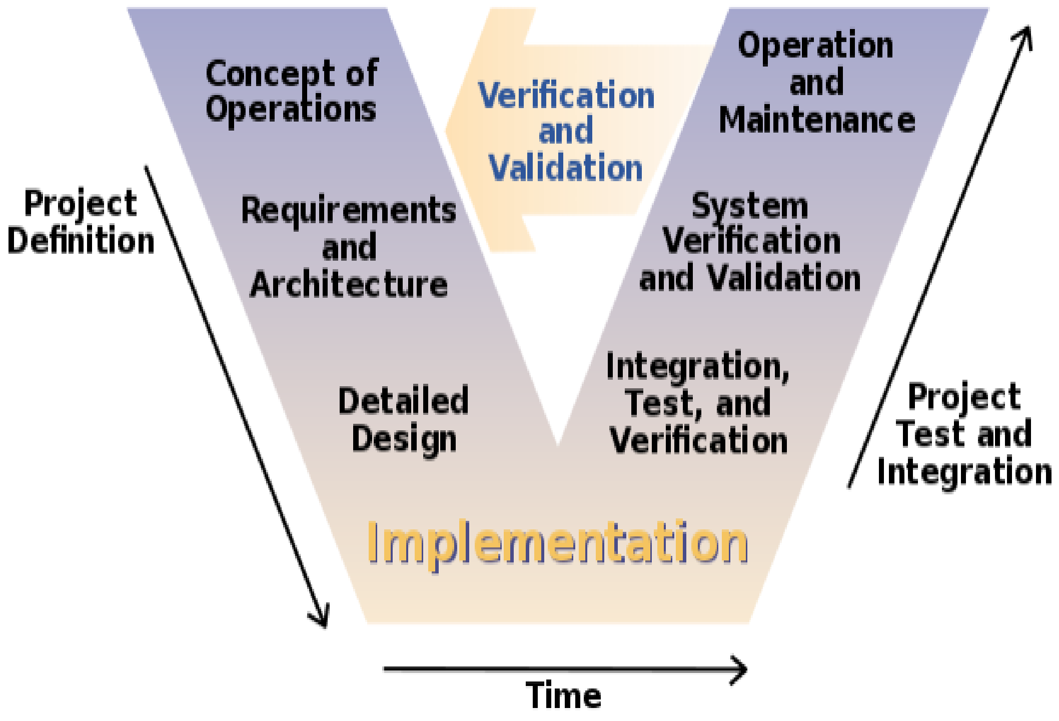
- The characteristics of incremental waterfall model
  - System development is broken down into many mini development projects
  - Partial systems are successively built to produce a final total system
  - Highest priority requirement is tackled first
- When to use incremental model
  - Requirements of the system are clearly understood
  - When demand for an early release of a product arises
  - When software engineering team are not very well skilled or trained
  - When high-risk features and goals are involved

Incremental waterfall model – cont.

Advantages	Disadvantages
The software will be generated quickly during the software life cycle	It requires a good planning designing
It is flexible and less expensive to change requirements and scope	Problems might cause due to system architecture as such not all requirements collected up front for the entire software lifecycle
Throughout the development stages changes can be done	Each iteration phase is rigid and does not overlap each other
This model is less costly compared to others	Rectifying a problem in one unit requires correction in all the units and consumes a lot of time
A customer can respond to each building	
Errors are easy to be identified	

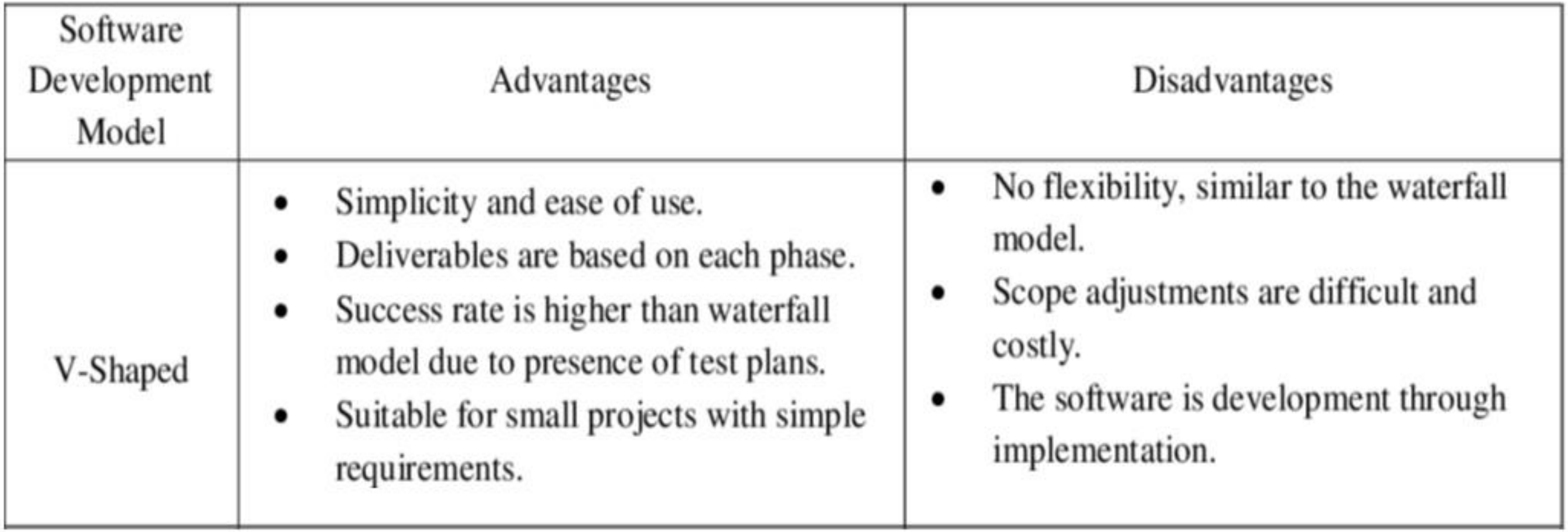
V-model

- V-model is a SDLC model where execution of processes happens in a sequential manner in a V-shape.
- It is also known as Verification and Validation model.
- It is considered an extension of waterfall model


Project Definition
Concept of Operations
Requirements and Architecture
Detailed Design
Verification and Validation
Operation and Maintenance
System Verification and Validation
Integration, Test, and Verification
Project Test and Integration
Implementation
Time

V-model – cont.

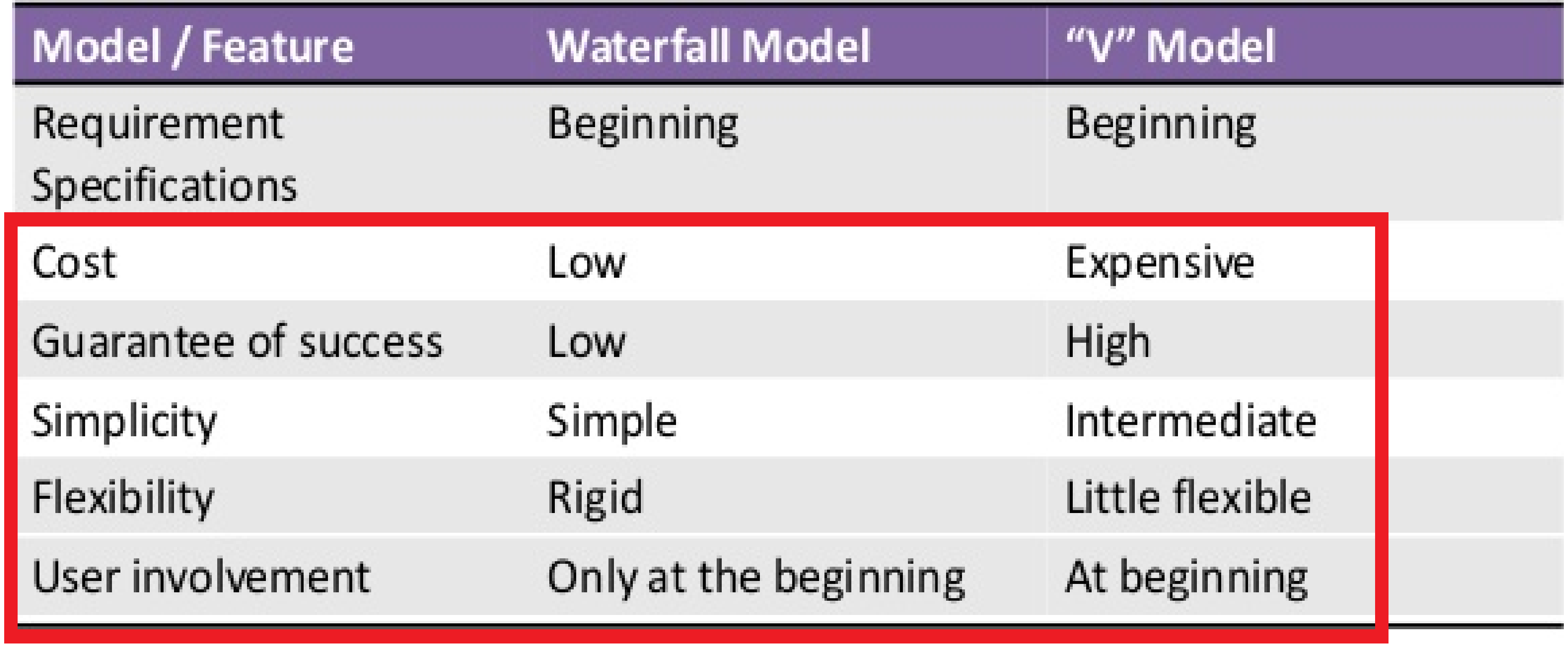
- The advantages and disadvantages of V-Model



Software Development Model	Advantages	Disadvantages
V-Shaped	• Simplicity and ease of use. • Deliverables are based on each phase. • Success rate is higher than waterfall model due to presence of test plans. • Suitable for small projects with simple requirements.	• No flexibility, similar to the waterfall model. • Scope adjustments are difficult and costly. • The software is development through implementation.

V-model – cont.

- V-Model vs. pure waterfall model

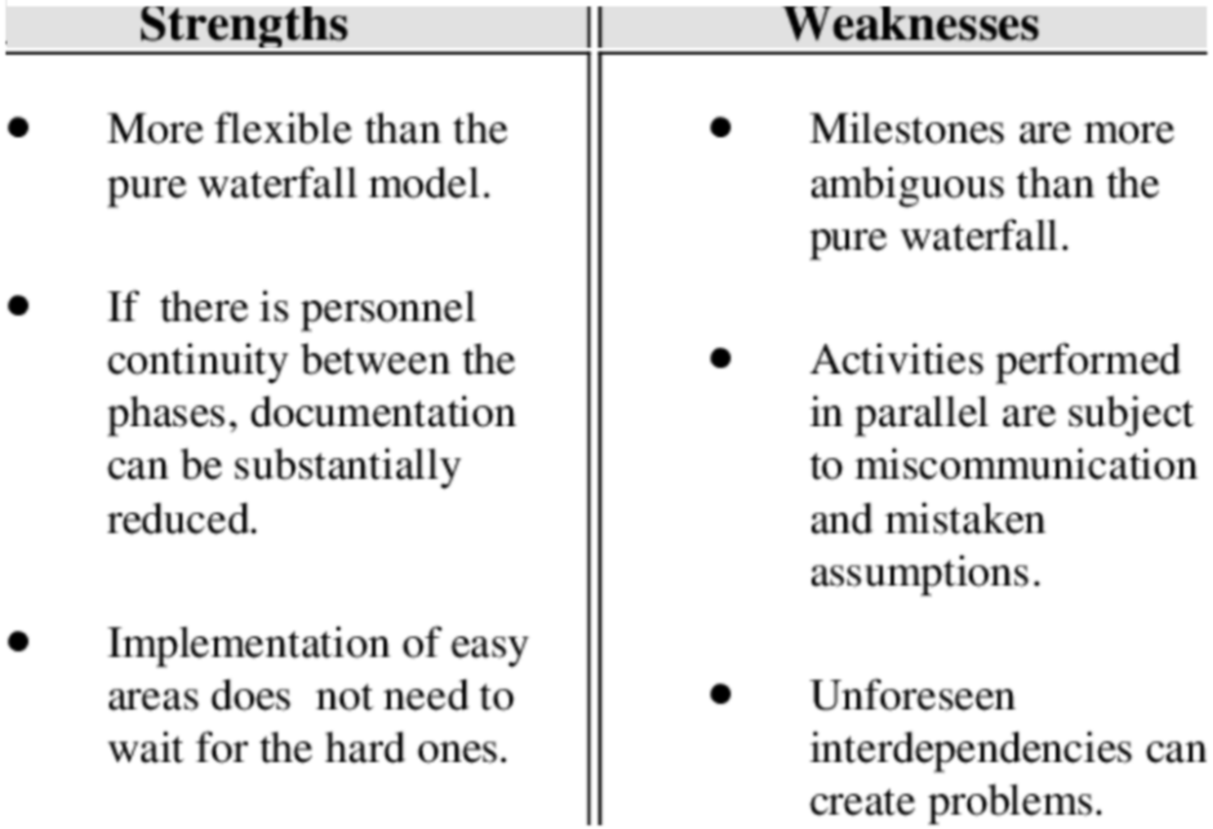


Model / Feature	Waterfall Model	"V" Model
Requirement Specifications	Beginning	Beginning
Cost	Low	Expensive
Guarantee of success	Low	High
Simplicity	Simple	Intermediate
Flexibility	Rigid	Little flexible
User involvement	Only at the beginning	At beginning

Waterfall model and its variants

- Waterfall model
- Waterfall with feedback model
- Sashimi model
  - (waterfall with overlap model)
- **Incremental waterfall model**
- V-model
  - (verification and validation model)

Modified waterfall models



Strengths	Weaknesses
• More flexible than the pure waterfall model.	• Milestones are more ambiguous than the pure waterfall.
• If there is personnel continuity between the phases, documentation can be substantially reduced.	• Activities performed in parallel are subject to miscommunication and mistaken assumptions.
• Implementation of easy areas does not need to wait for the hard ones.	• Unforeseen interdependencies can create problems.

Summary

- Predictive models vs. adaptive models
- Waterfall model
  - Concept `
  - Advantages and disadvantages
- Waterfall with feedback model
  - Concept
- Sashimi model (waterfall with overlapping phase)
- Incremental waterfall model
  - Concept
- V-model
  - Concept
- Comparison

Announcement

- Test 2 next Tuesday (Nov. 12), cover week 5-8 topics. Please bring your laptop
  - Locations
    - BA208 (the first letter of your family name from A-H (42))
    - BA211 (the first letter of your family name from I-P (31))
    - BA112 (the first letter of your family name from Q-Z (28))
- Group project
  - Software design document – project report due date Nov. 22