



Software Development Management

Lecture 3

Software Development Models

By: Hossein Momeni

momeni@iust.ac.ir

Fall 2011

Mazandaran University of Science and Technology



Overview

- Introduction
- Technical plan
- Software process models
- Selecting process model



Definitions

- Software Process
 - the set of activities, methods, and practices that are used in the production and evolution of software
- Software Process Model
 - one specific embodiment of a software process architecture



Why Modelling?

- To provide a common understanding
- To locate any inconsistencies, redundancies and omissions
- To reflect the development goals and provide early evaluation
- To assist development team to understand any special situation



Project Analysis

- Methodologies
 - Object-Oriented Development (OOD)
 - Structured System Analysis and Design Method (SSADM)
 - Jackson Structured Programming (JSP)
- Technologies
 - application-building environment
 - knowledge-based system tools



Project Characteristics

- data oriented or control oriented system?
- general package or application specific?
- a particular type of system for which specific tools have been developed?
- safety-critical system?
- nature of the hardware/software environment?



Project Risks

- Product uncertainty
- Process uncertainty
- Resource uncertainty



Considerations for Project Approach

- Control systems
- Information systems
- General applications
- Specialized techniques
- Hardware environment
- Safety-critical systems
- Imprecise requirements



Technical Plan

- Contents
 - Constraints
 - Approach
 - Implementation
 - Implications



Technical Plan - Constraints

- Character of the system to be developed
- Risks and uncertainties of the project
- User requirements concerning implementation



Technical Plan - Approach

- Selected methodology or process model
- Development methods
- Required software tools
- Target hardware/software environment



Technical Plan - Implementation

- Development environment
- Maintenance environment
- Training



Technical Plan - Implications

- Project products and activities
 - effect on schedule duration and overall project effort
- Financial
 - report used to produce costings



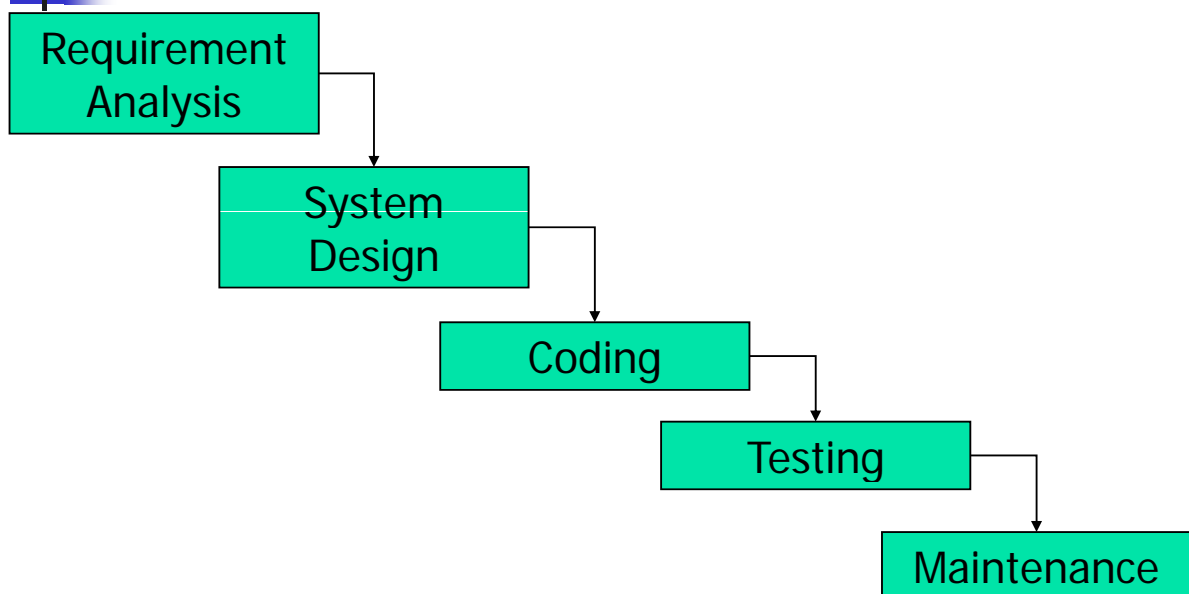
Software Process Models

- Waterfall Model
- V Model
- Spiral Model
- Prototyping Model

Software Process Models (cont'd)

- Phased Development Model
 - incremental development model
 - iterative development model
- Operational Specification Model
- Transformation Model

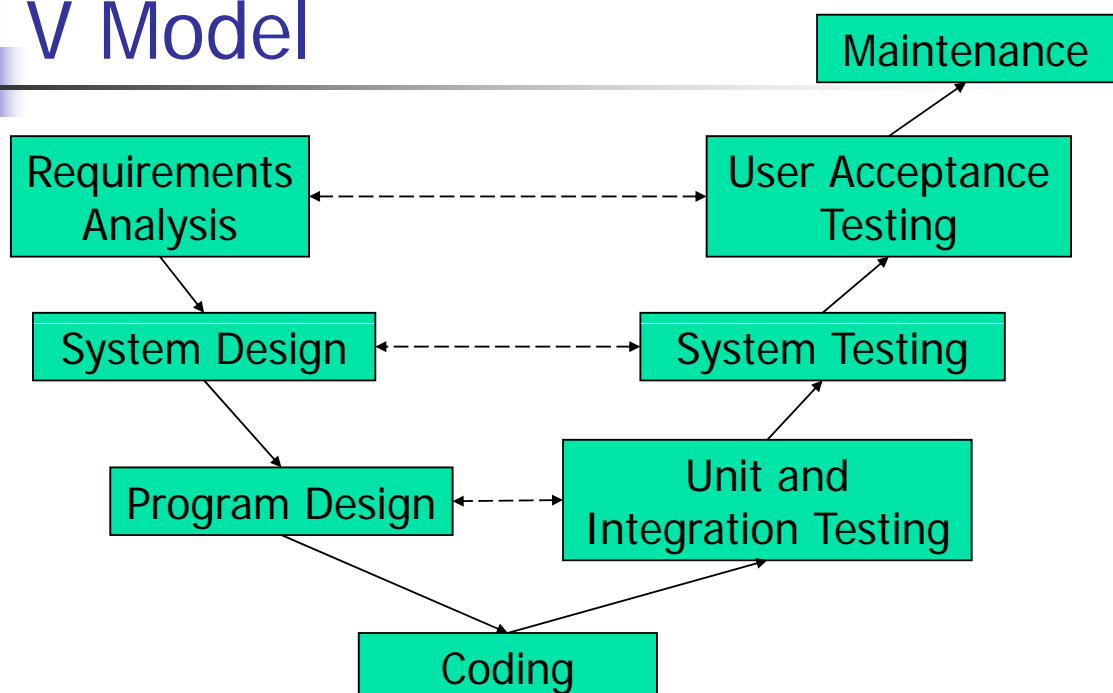
Waterfall Model



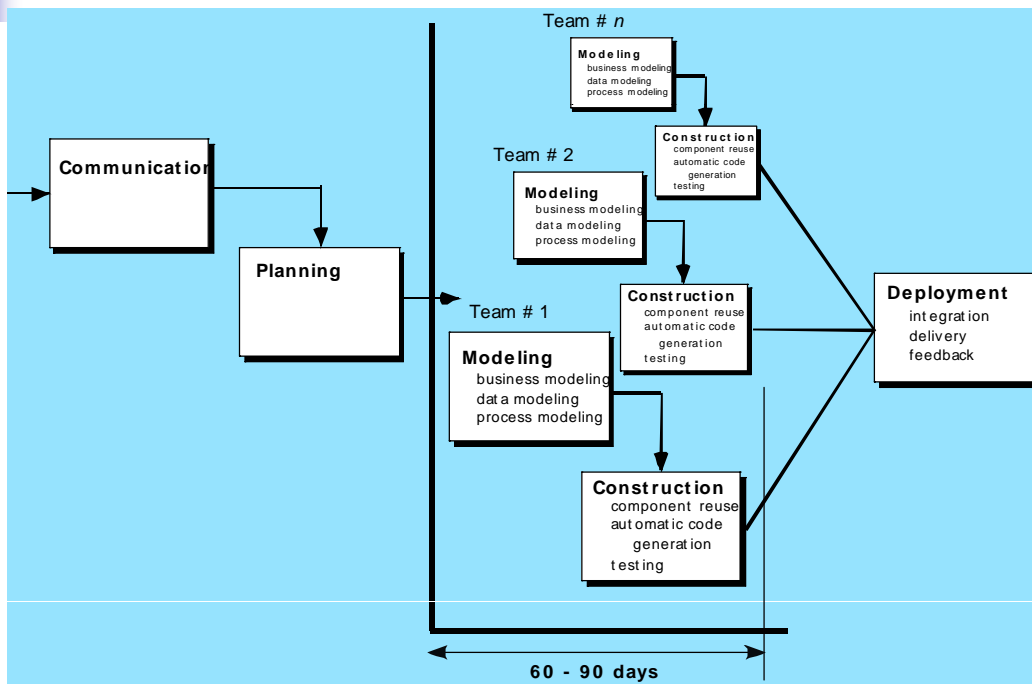
Waterfall Model (cont'd)

- classical
- one-shot approach
- effective control
- limited scope of iteration
- long cycle time
- not suitable for system of high uncertainty

V Model



The RAD Model

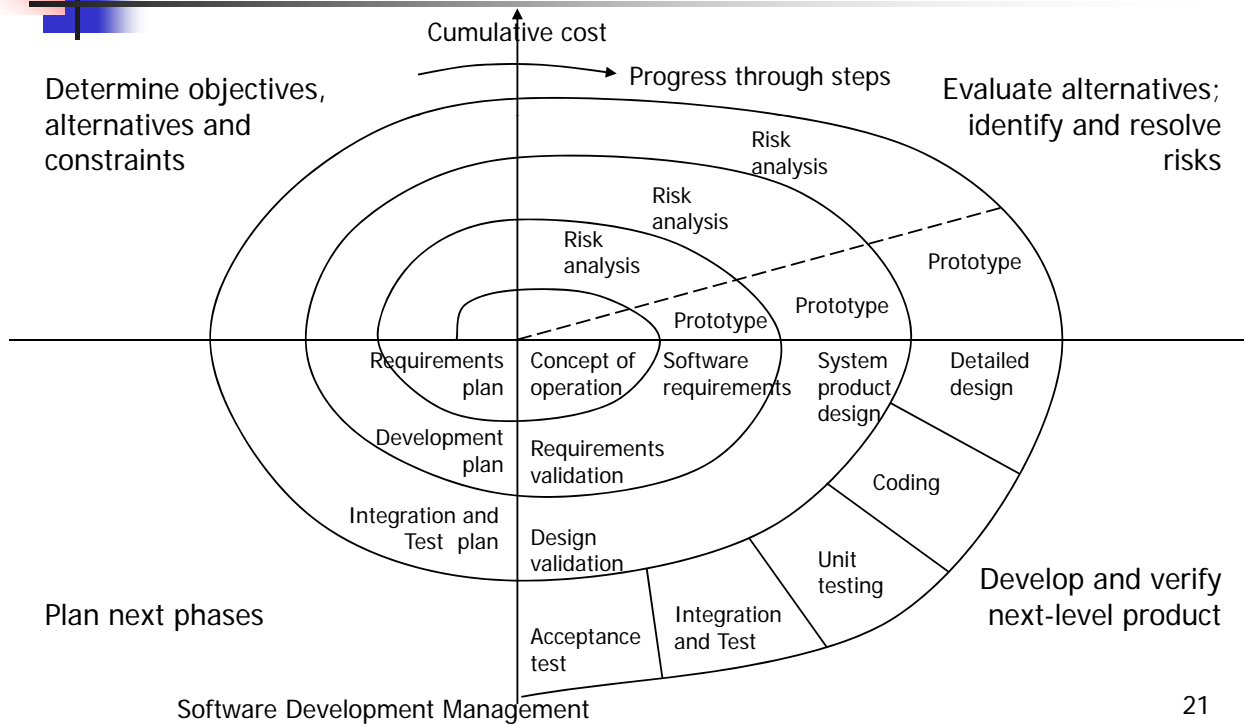


19

V Model (cont'd)

- Additional validation process introduced
- Relate testing to analysis and design
- Loop back in case of discrepancy

Spiral Model (adapted from Boehm 1987)



21

Spiral Model (cont'd)

- Evolutionary approach
- Iterative development combined with risk management
- Risk analysis results in "go, no-go" decision



Spiral Model (cont'd)

- Four major activities
 - Planning
 - Risk analysis
 - Engineering
 - Customer evaluation



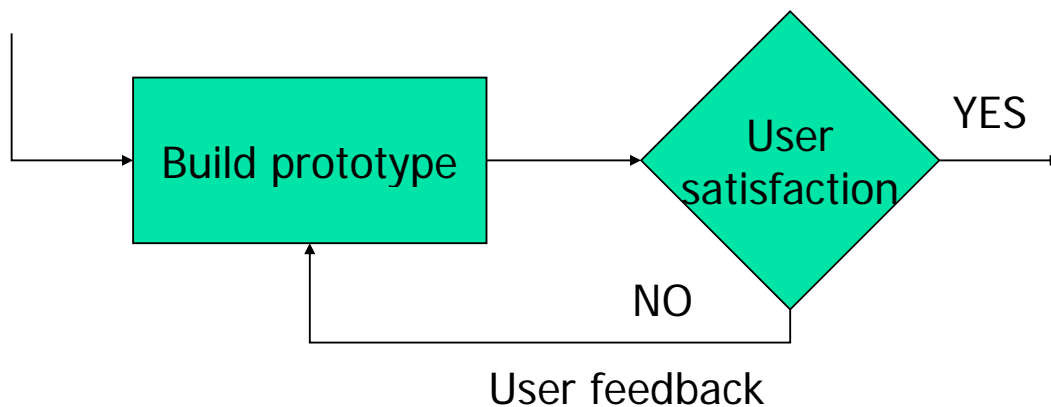
Prototyping Model

- Goals
 - meet users' requirements in early stage
 - reduce risk and uncertainty

Classification of Prototype

- **Throw-away**
 - After users agree the requirements of the system, the prototype will be discarded.
- **Evolutionary**
 - Modifications are based on the existing prototype.
- **Incremental**
 - Functions will be arranged and built accordingly.

Prototyping Model





Benefits of Prototyping

- Learning by doing
- Improved communication
- Improved user involvement
- Clarification of partially-known requirements



Prototyping Sequences

- Requirements gathering
- Quick design
- Prototype construction
- Customer evaluation
- Refinement
- Loop back to quick design for fine tuning
- Product engineering



Benefits of Prototyping

- Demonstration of the consistency and completeness of a specification
- Reduced need for documentation
- Reduced maintenance costs
- Production of expected results



Drawbacks of Prototyping

- Users sometimes misunderstand the role of the prototype
- Lack of project standards possible
- Lack of control
- Additional expense



Prototype Products

- Human-computer interface
- System functionality



Prototype Changes

- Three categories
 - Cosmetic (35%)
 - screen layout
 - Local (60%)
 - screen processing
 - Global (5%)
 - multi-parts processing
 - design review



Phased Development

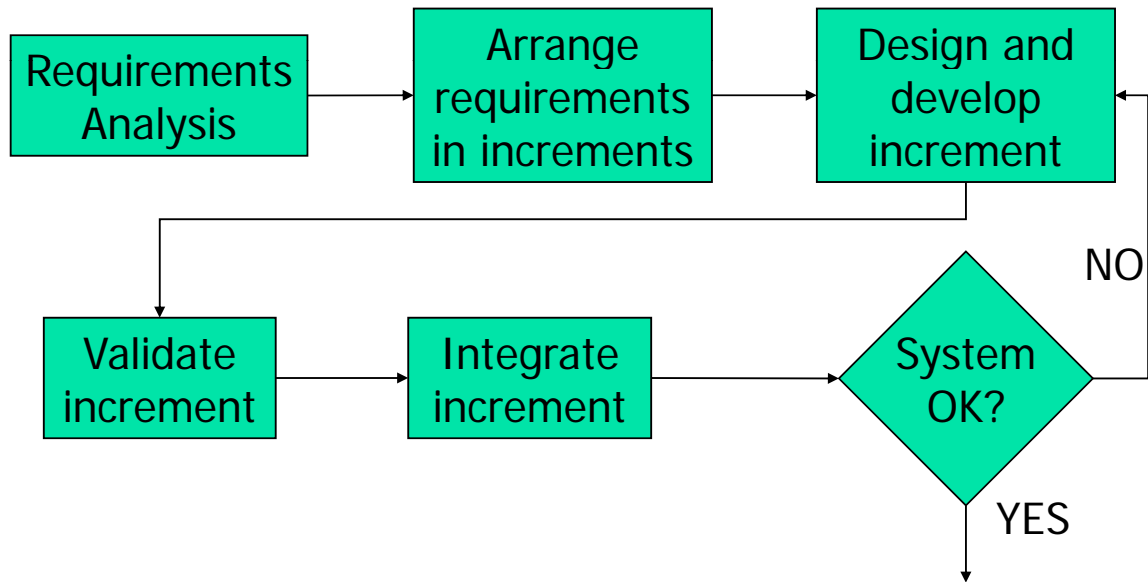
- Reduce cycle time
- Two parallel systems:
 - operational system (Release n)
 - development system (Release n+1)
- Two approaches
 - incremental
 - iterative



Incremental Model

- Break system into small components
- Implement and deliver small components in sequence
- Every delivered component provides **extra functionality** to user

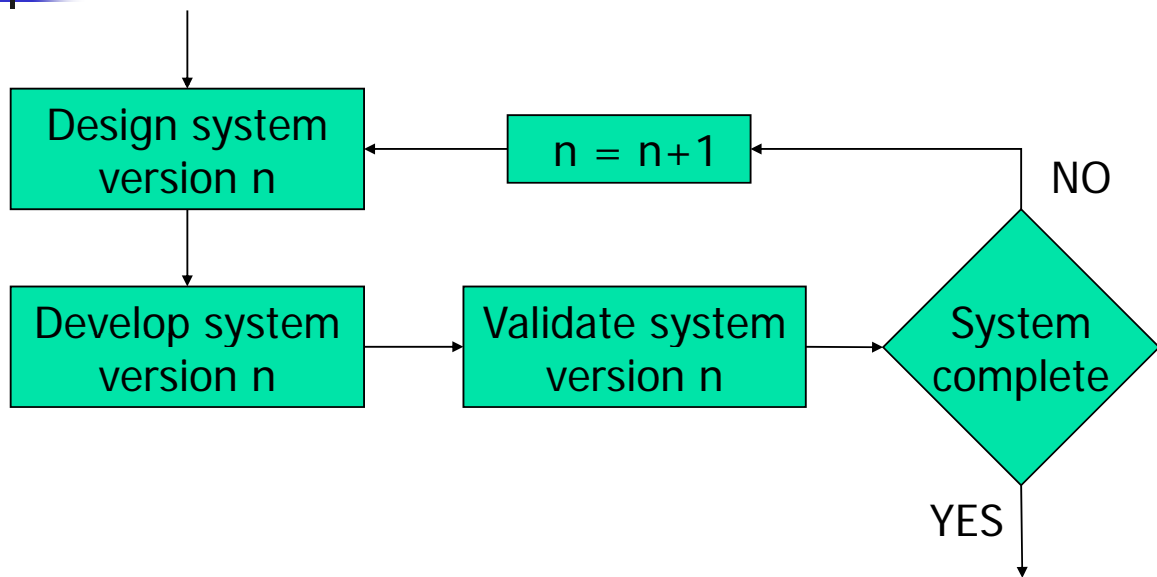
Incremental Model (cont'd)



Iterative Model

- Deliver full system in the beginning
- Enhance functionality in new releases

Iterative Model (cont'd)



Combined Incremental and Iterative Model

- Every new release includes
 - extra functionality
 - enhancement of existing functionality
- Popularly used in software industry



Advantages of Phased Development

- Early feedbacks
- Less possible requirement changes
- Early benefits for users
- Easier control and manage



Advantages of Phased Development (cont'd)

- Capture early market
- Facilitate early training
- Can be temporarily abandoned
- Increase job satisfaction



Disadvantages of Phased Development

- 'Software breakage'
- Reduced productivity



Operational Specification Model

- Use executables to demonstrate system behaviour
- Resolve requirement uncertainties in early stage
- Merging functionality and system design



Transformational Model

- Transform a specification into delivered system
- Require automated support
- Rely on formal specification method