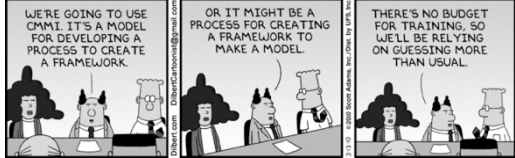


**CIS 422/522**

Software Life cycles and Process models  
Team Assignments & First Meeting



CIS 422/522 Fall 2012 1

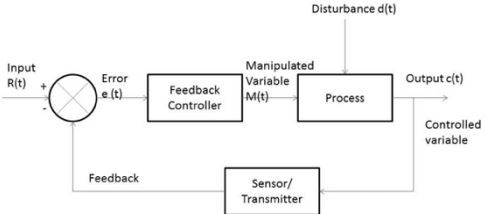
**View of SE in this Course**

- The purpose of software engineering is to *gain and maintain* intellectual and managerial control over the *products and processes* of software development.
- Intellectual control implies
  - We understand the developmental goals
  - Can distinguish good choices from bad
  - We can effectively build to meet our goals
    - Behavioral requirements (functionality)
    - Software Qualities (reliability, security, maintainability, etc.)
- Managerial control implies
  - We make accurate recourse estimates
  - We deliver on schedule and within budget

CIS 422/522 Fall 2012 2

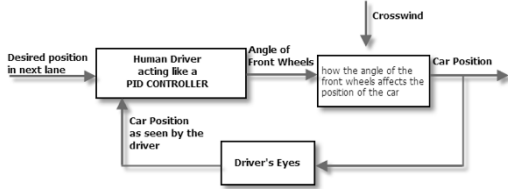
**Control Realities**

- Reality Check:
  - Cannot fully predict consequences of our choices
  - Control is never absolute
- Implication: maintaining control is an active process (view as a feedback-control loop)



CIS 422/522 Fall 2012 3

**Active Control**

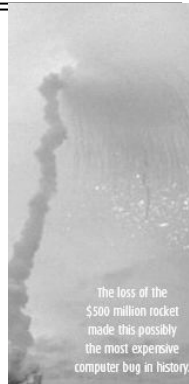


- Control in a software development means
  - Understand where we want to be (ideal)
  - Measure current delta
  - Make adjustments

CIS 422/522 Fall 2012 4

## Control and Risk

- Risk: a *risk* is defined as a condition that can lead to a loss of control
  - Incorrect, misunderstood, or missing requirements
  - Poor design choices
  - Poorly matched interfaces
  - Differing assumptions by developers
  - Inadequate testing, validation, etc.
- Can lead to delivering wrong product, late, over cost..
- Well defined processes help organize work and control risks



CIS 422/522 Fall 2012

5

## Need to Organize the Work

- Nature of a software project
  - Software development produces a set of interlocking, interdependent work products
    - E.g. Requirements -> Design -> Code -> Test
  - Implies dependencies between tasks
  - Implies dependencies between people
- Must organize the work such that:
  - Every task gets done
  - Tasks get done in the right order
  - Tasks are done by the right people
  - The product has the desired qualities
  - The end product is produced on time

CIS 422/522 Fall 2012

6

## Addressed by Software Processes

- Developed as a tool for controlling complex software developments
- Answers the “who”, “what”, “when”, etc. questions
  - What product should we work on next?
  - What kind of person should do the work?
  - What information is needed to do the work?
  - When is the work finished?
- Intended use
  1. *Model* of development (what does or should occur)
  2. *Guide* to developers in what to produce and when to produce it

CIS 422/522 Fall 2012

7

## Definitions

- *Software Life Cycle*: evolution of a software development effort from concept to retirement
- *\*Software Process Model*: Abstract representation of a software life cycle as a set of
  1. Activities: tasks to be performed (how)
  2. Artifacts; work products produced (what)
  3. Roles: skills needed (who)
- *Software Process*: institutionalized version of a life software model defining specific roles, activities, and artifacts

CIS 422/522 Fall 2012

8

---

## Common Process Models

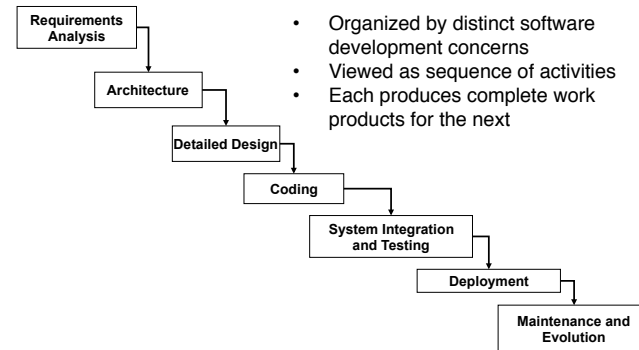
Waterfall  
Prototyping  
Iterative  
Spiral  
Agile

CIS 422/522 Fall 2012

9

---

## A "Waterfall" Model



CIS 422/522 Fall 2012

10

---

## Activities and Products

- **Requirements Analysis**
  - Goal: understand and define what the software must do and any properties it must have
  - Product: Software Requirements Specification (SRS)
  - Role: Requirements Analyst
- **Architectural Design**
  - Goal: decompose of the problem into components that together satisfy the requirements
  - Products: architectural design specification, interface specs.
  - Role: Software Architect
- **Detail Design**
  - Goal: internal design of components (e.g., objects) defining algorithms and data structures supporting the interface
  - Products: design documentation, pseudo-code
  - Role: Coder

CIS 422/522 Fall 2012

11

---

## Phases and Products

- **Implementation**
  - Goal: realization of the design in machine-executable form
  - Product: code
  - Role: Coder
- **Integration and Testing**
  - Goal: validation and verification of the implementation against requirements and design
  - Products: test plan, test cases
  - Role: tester
- **Maintenance (not really one activity)**
  - Goal: repair errors or update deployed system
  - Products: bug fixes, patches, new versions
  - Role: Architect, Coder, Tester

CIS 422/522 Fall 2012

12

### Waterfall Model Variations

---

There have been many variations

CIS 422/522 Fall 2012 13

### A “Waterfall” Model

---

What are the issues:

1. What kinds of risks are addressed?
2. What kinds of risks are not addressed?

CIS 422/522 Fall 2012 14

### A “Waterfall” Model\*

---

As a guide: does not address common development risks

- What happens if requirements are wrong?
- If scheduling or budget is wrong?

Problems of temporal Distance\*

\*not in text CIS 422/522 Fall 2012 15

### Characteristic Model: Prototyping

---

- Waterfall variation
- First system versions are prototypes, either:
  - Interface
  - Functional
- Which waterfall risks does this try to address?

CIS 422/522 Fall 2012 16

### Characteristic Processes: The Iterative Model

---

- Process viewed as a sequence of iterations
  - Essentially, a *series of waterfalls*
  - Each iteration builds on the previous one (e.g., adds requirements, design components, code features, tests)
  - Each iteration produces complete set of work products including deliverable software

CIS 422/522 Fall 2012 17

### Iterative Model

---

- Also called “incremental development”
- Addresses some common waterfall risks
  - Risk that software cannot be completed – build incremental subsets
  - Risk of building the wrong system – stakeholder have opportunities to see the software
  - Also, feasibility, schedule, budget and others to some extent

CIS 422/522 Fall 2012 18

### Characteristic Processes: The Spiral Model

---

- Process viewed as repeating cycles of increasing scale
- Identify risks and determine (next set of) requirements
- Each cycle builds next version by extension, increasing scale each time

CIS 422/522 Fall 2012 19

### Spiral Model

---

CIS 422/522 Fall 2012 20

## Spiral Model Goals

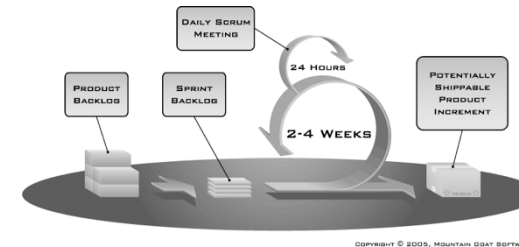
---

- Response lack of risk analysis and risk mitigation in “waterfall” process
- Explicit risk analysis and mitigation at each phase (e.g., prototyping)
- Explicit Go/No-Go decision points in process

## Characteristic Processes: Agile (e.g. scrum)

---

- Process viewed as nested sequence of builds (sprints)
  - Each build adds small feature set (one or two)
  - Nightly build/test, frequent customer validation
  - Focus on delivering code, little or no time spent on documentation



## Also...

---

- RAD models
- Extreme Programming
- Etc., etc.

Why so many models?

## How do we Choose a Development Process?

---

E.g., for your projects

## Objectives

- Goal: proceed as rationally and systematically as possible (i.e., in a controlled manner) from a statement of goals to a design that demonstrably meets those goals within design and management constraints
  - Understand that any process description is an abstraction
  - Always must compensate for deviation from the ideal (e.g., by iteration)
  - Still important to have a well-defined process to follow and measure against

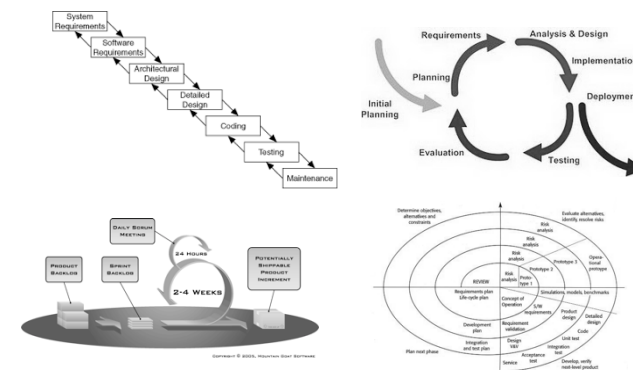
## A Software Engineering Perspective

- Question of control vs. cost
- Choose processes, methods, notations, etc. to provide an appropriate level of control for the given product and context
  - Sufficient control to achieve results
  - No more than necessary to contain cost and effort
- Provides a basis for choosing or evaluating processes, methods, etc.
  - Does it achieve our objectives at reasonable cost?
  - Does it address the most important developmental risks?

## Project Relevance

- Need to agree on kind of control you need and how you will accomplish it
- Process model will then help keep everyone on track
  - Basis for planning and scheduling
  - Each person knows what to do next
  - Basis for tracking progress against schedule
- Should be one of the first products you produce but expect it to evolve

## Exercise: Which Model?



## Exercise: Project Processes

---

- Discuss: which process is the best fit for your projects and why?
- For each process you do not select, what characteristics do not fit well with the project
- For the process selected
  - How does it fit with project characteristics?
  - How does it help address project risks?

## Project Preparation

---

Teams  
Worksite

## Take-away

---

- Expected to know standard processes and their rationale
- Understand how and why people use different development models
- Understand how to choose an appropriate model for a given developments

## Assignment

---

- Project
  - Forward your emails from xxx@uoregon.edu
  - First meeting (in class)
  - Plan and hold at least one project meeting out of class
    - Choose a team name
    - Create team *assembla* page on wiki
  - Record meeting notes (Meeting Notes page)
  - Fill out Develop Logs
  - Monday:
    - Status report on your decisions from each team
    - Show completed team page



## Team Assignments

---

- Team Assignments
- Assembla walkthrough

---

Questions?