

CIS 422/522

Software Processes Part 2

- Process Models
- Choosing a Process
- Project workspaces/start



CIS 422/522 © S. Faulk

1

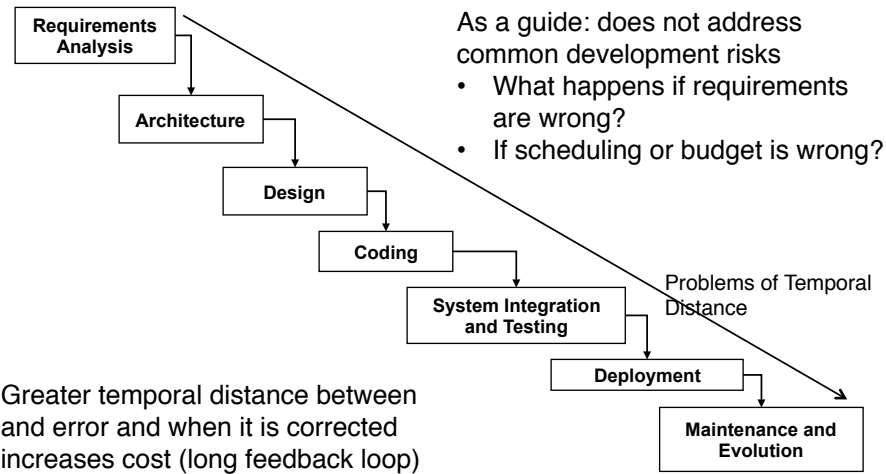
Software Processes

- Developed as a conceptual tool for organizing complex software developments
- Organize the work
- Address developmental risks
- Intended use (idealized)
 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 © S. Faulk

2

A "Waterfall" Model*

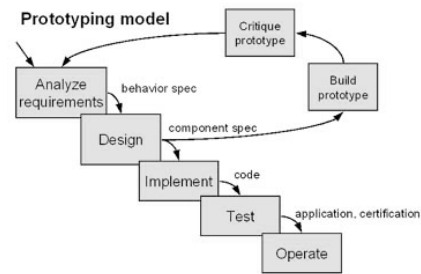


Common Process Models

Waterfall
 Prototyping
 Iterative
 Spiral
 Agile

Characteristic Model: Prototyping

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

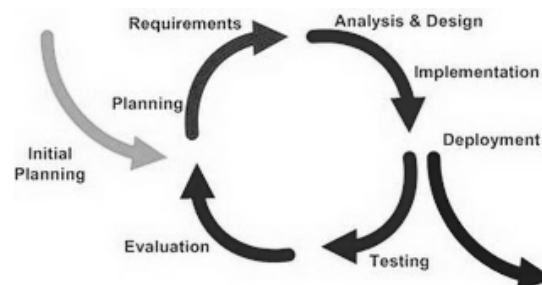


CIS 422/522 © S. Faulk

5

Characteristic Processes: The Iterative Model

- Process is 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 deliverable software
 - Customers provide feedback on each release
 - There is no “maintenance” phase – each version includes problem fixes as well as new features



6

Iterative Model

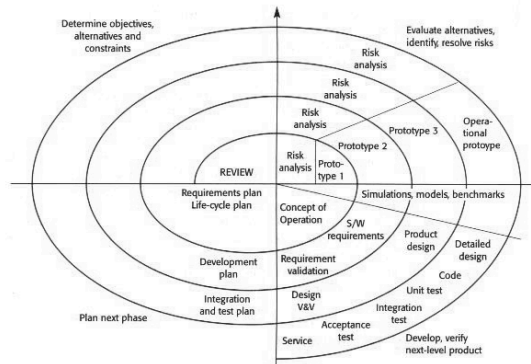
- Also called “incremental development”
- Addresses some common waterfall risks
 - Risk that software cannot be completed – build incremental subsets
 - Risk of building the wrong system – stakeholder have opportunities to see the software each increment
 - Each iteration provides checkpoint for feasibility,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

Advantages of Incremental Development

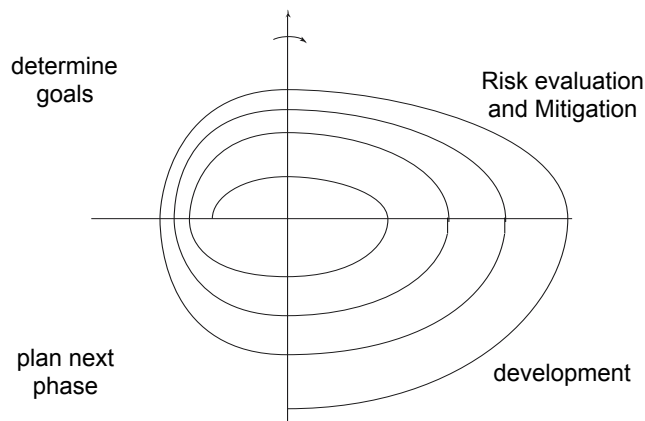
- Customers get usable functionality earlier than with waterfall
- Early feedback improves likelihood of producing a product that satisfies customers
 - Reduces market risk: if customers hate the product, find out before investing too much effort and money
- The quality of the final product is better
 - The core functionality is developed early and tested multiple times
 - Only a relatively small subset of functionality added in each release: easier to get it right and test it thoroughly
 - Detect design problems early and get a chance to redesign

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



Spiral Model



Spiral Model Characteristics

- Response lack of explicit risk analysis and risk mitigation in “waterfall” process
- Includes risk analysis and mitigation activities at each phase (e.g., prototyping)
- Explicit Go/No-Go decision points in process
- “Heavy-weight” process: substantial overhead not contributing directly to end products

Characteristic Processes: Agile (e.g. scrum)

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



How do we Choose a Development Process?

E.g., for your projects

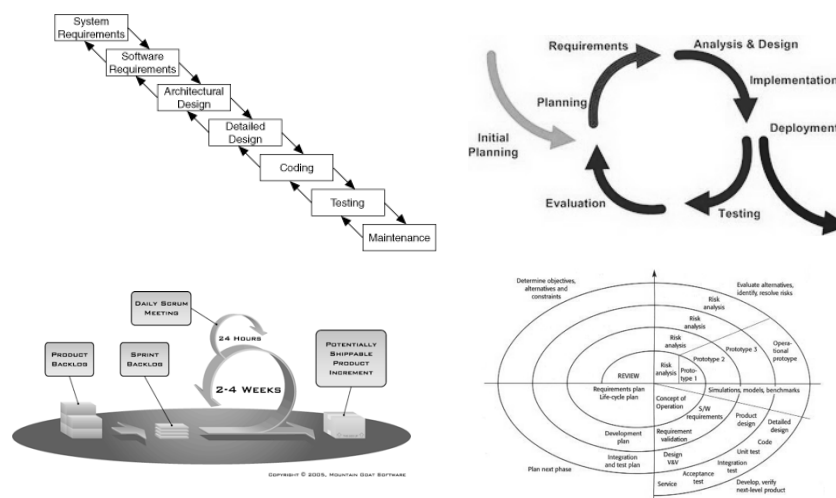
Objectives

- 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: processes introduce *overhead*
- Choose process to provide an appropriate level of control for the given product and context
 - Sufficient control to mitigate risks, 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?
- Need to agree on kind of control you need and how you will accomplish it

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?

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
 - Often poorly understood in industry

Project Preparation

Project Requirements
Assembla Worksites

Project

- **Goal: be clear on what you plan to build**
 - Are the project requirements complete and well defined? If not, what will you do about it?
 - Clarify Address Book requirements
 - Generate questions for instructor
- **Assembla: start drafting a project plan**
 - Understand how to edit the Wiki
 - Put up team name and logo
 - Identify team roles (even if they might change)
 - Lay out initial cut at schedule (using the class Schedule)
 - Create first meeting notes, developer logs

Schedule

- Monday: “stand-up” meeting. Each team gives a 2-3 minute summary of progress against plan (see “Deliverables” on Schedule page)
 - What was accomplished
 - What is planned for the week
 - Any problems or obstacles to progress

Questions?