Human Computer Interaction (INSY4112)

CHAPTER 5

Interaction Design and HCI in the Software Process

HCI IN THE SOFTWAREPROCESS

- Introduction
- The software lifecycle
- Usability engineering
- Interactive design and prototyping
- Design rationale

Objectives

- To Produce a low-fidelity prototype for an interactive product based upon a simple list of interaction design principles.
- To Communicate effectively to peers and specialists about requirements, design, and evaluation activities relating to interactive software products

HCI in the Software Process

➢ Introduction on Software Process

- **Software engineering** provides a means of understanding the structure of the design process, and that process can be assessed for its effectiveness in interactive system design.
- **Usability engineering** promotes the use of explicit criteria to judge the success of a product in terms of its usability.

HCI in the Software Process

➢ Introduction on Software Process

- **Iterative design** practices work to incorporate crucial customer feedback early in the design process to inform critical decisions which affect usability.
- Design involves making many decisions among numerous alternatives. **Design rationale** provides an explicit means of recording those design decisions and the context in which the decisions were made.

HCI in the Software Process

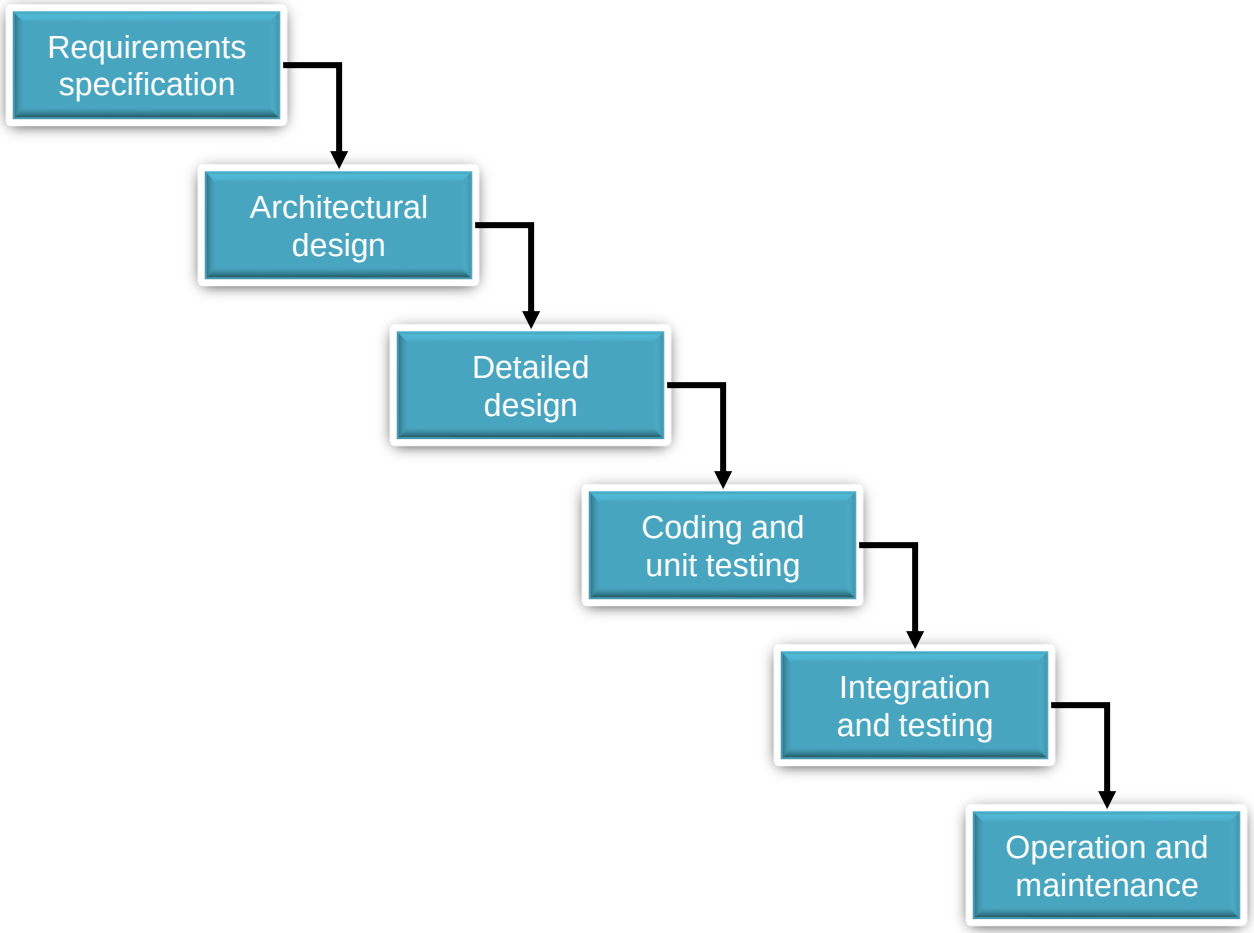
➢ The Software Lifecycle

- Software engineering is the discipline for understanding the software design process, or life cycle

- Designing for usability occurs at all stages of the life cycle, not as a single isolated activity

HCI in the Software Process

➢ The waterfall model



HCI in the Software Process

➢ Activities in the life cycle

❖ Requirements specification

- ✓ Designer and customer try capture what the system is expected to provide can be expressed in natural language or more precise languages, such as a task analysis would provide

HCI in the Software Process

➢ Activities in the life cycle

❖ Architectural design

- ✓ High-level description of how the system will provide the services required factor system into major components of the system and how they are interrelated needs to satisfy both functional and nonfunctional requirements

HCI in the Software Process

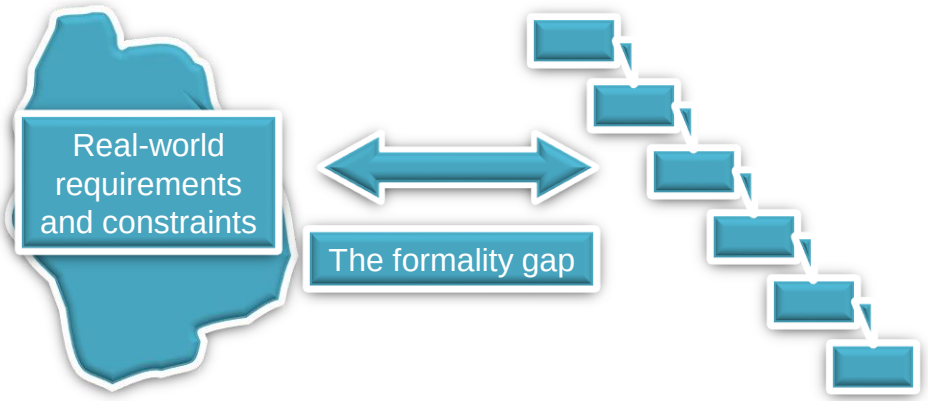
➢ Activities in the life cycle

❖ Detailed design

- ✓ Refinement of architectural components and interrelations to identify modules to be implemented separately the refinement is governed by the non-functional requirements

HCI in the Software Process

➢ Verification and validation

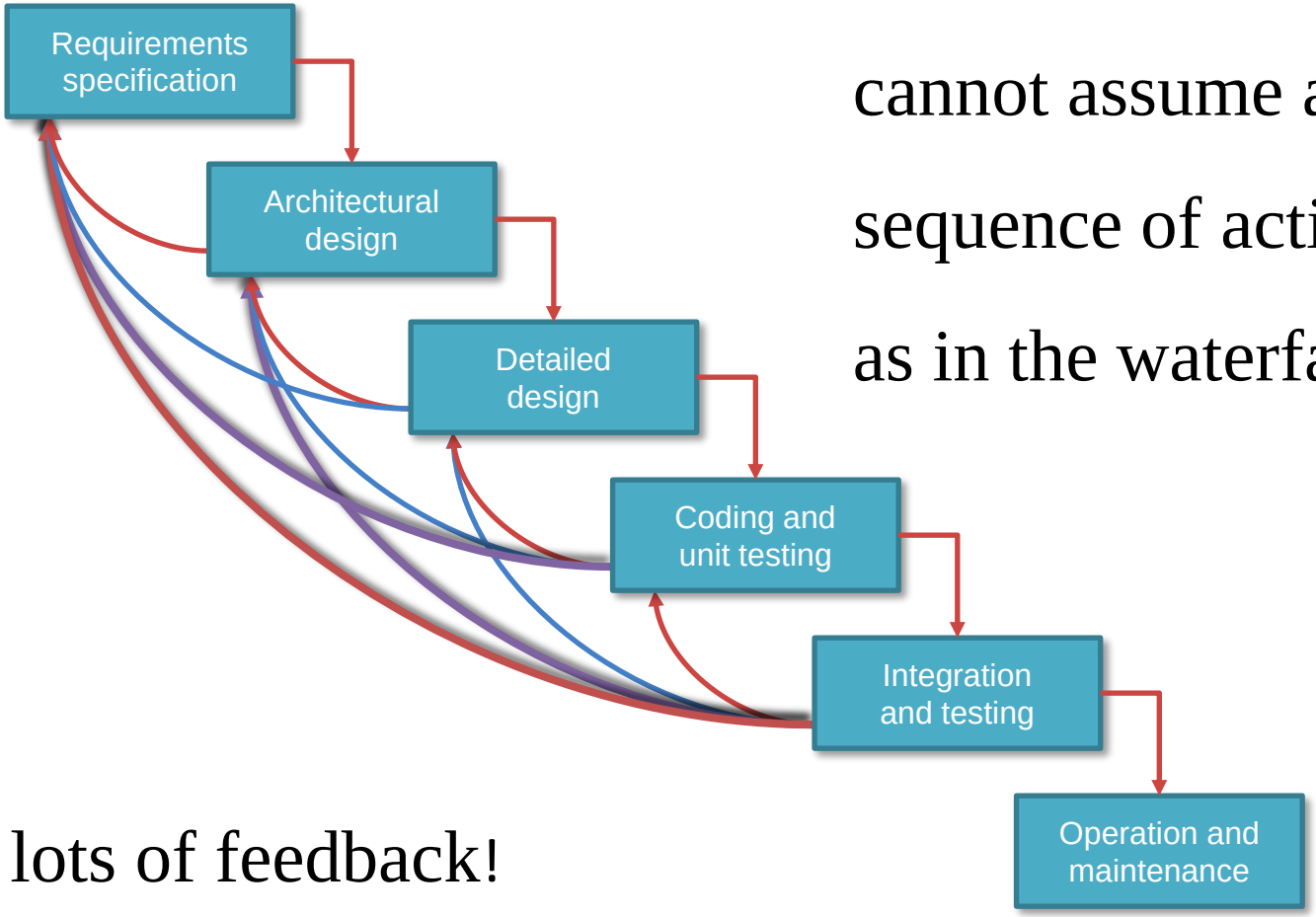


- Verification
  - ✓ designing the product right
- Validation
  - ✓ designing the right product

- The formality gap
  - ✓ validation will always rely to some extent on subjective means of proof
- Management and contractual issues
  - ✓ design in commercial and legal contexts

HCI in the Software Process

➢ The life cycle for interactive systems



cannot assume a linear sequence of activities as in the waterfall model

lots of feedback!

HCI in the Software Process

➢ Usability engineering

- The ultimate test of usability based on measurement of user experience
- Usability engineering demands that specific usability measures be made explicit as requirements
- Usability specification
  - usability attribute/principle
  - measuring concept
  - measuring method
  - now level/ worst case/ planned level/ best case
- Problems
  - usability specification requires level of detail that may not be
  - possible early in design satisfying a usability specification
  - does not necessarily satisfy usability

HCI in the Software Process

- **part of a usability specification for a VCR**

Attribute: Backward recoverability	
Measuring concept:	Undo an erroneous programming sequence
Measuring method:	Number of explicit user actions to undo current program
Now level:	No current product allows such an undo
Worst case:	As many actions as it takes to program-in mistake
Planned level:	A maximum of two explicit user actions
Best case:	One explicit cancel action

HCI in the Software Process

➢ ISO usability standard 9241

❖ Adopts traditional usability categories:

- Effectiveness
  - ✓ can you achieve what you want to?
- Efficiency
  - ✓ can you do it without wasting effort?
- Satisfaction
  - ✓ do you enjoy the process?

HCI in the Software Process

➢ some metrics from ISO 9241

Usability objective	Effectiveness measures	Efficiency measures	Satisfaction measures
Suitability for the task	Percentage of goals achieved	Time to complete a task	Rating scale for satisfaction
Appropriate for trained users	Number of power features used	Relative efficiency compared with an expert user	Rating scale for satisfaction with power features
Learnability	Percentage of functions learned	Time to learn criterion	Rating scale for ease of learning
Error tolerance	Percentage of errors corrected successfully	Time spent on correcting errors	Rating scale for error handling

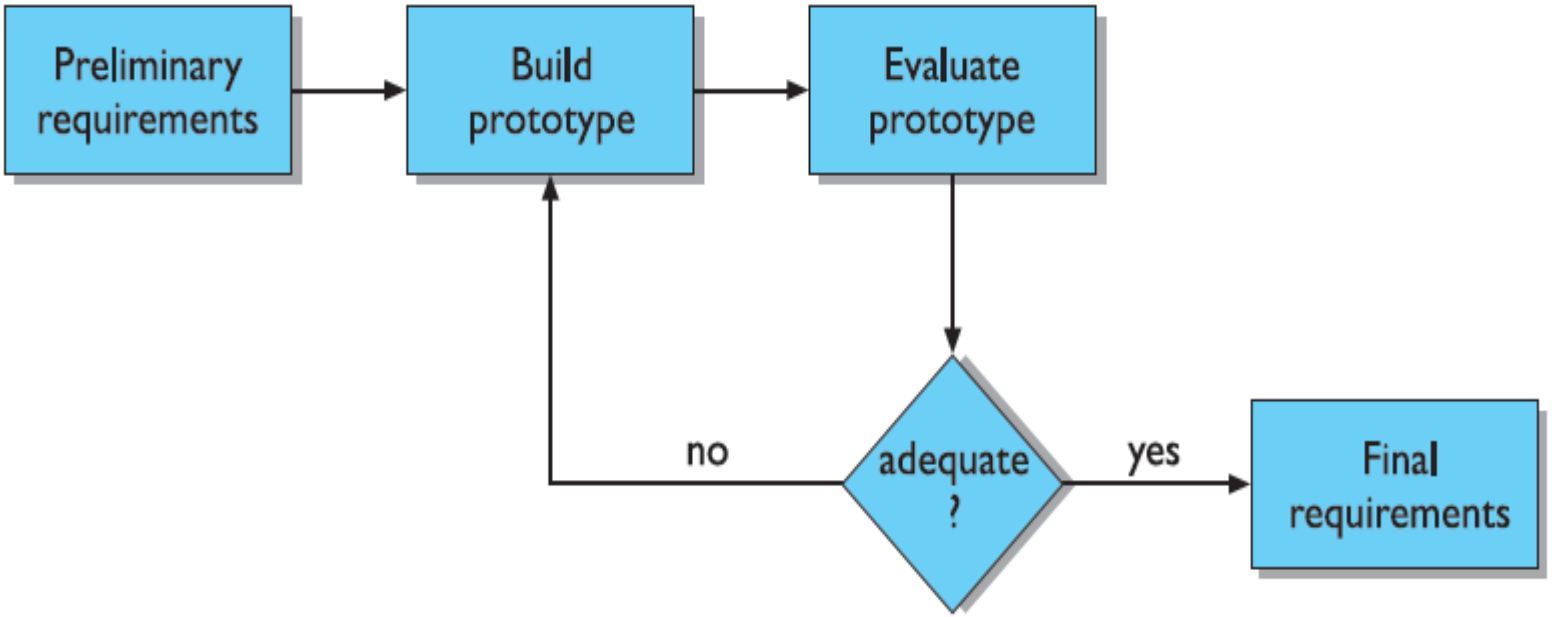
HCI in the Software Process

Iterative design and prototyping

- Iterative design overcomes inherent problems of incomplete requirements
- Prototypes
  - simulate or animate some features of intended system
  - different types of prototypes
    - throw-away
    - incremental
    - Evolutionary
- Management issues
  - time
  - planning
  - non-functional features
  - contracts

HCI in the Software Process

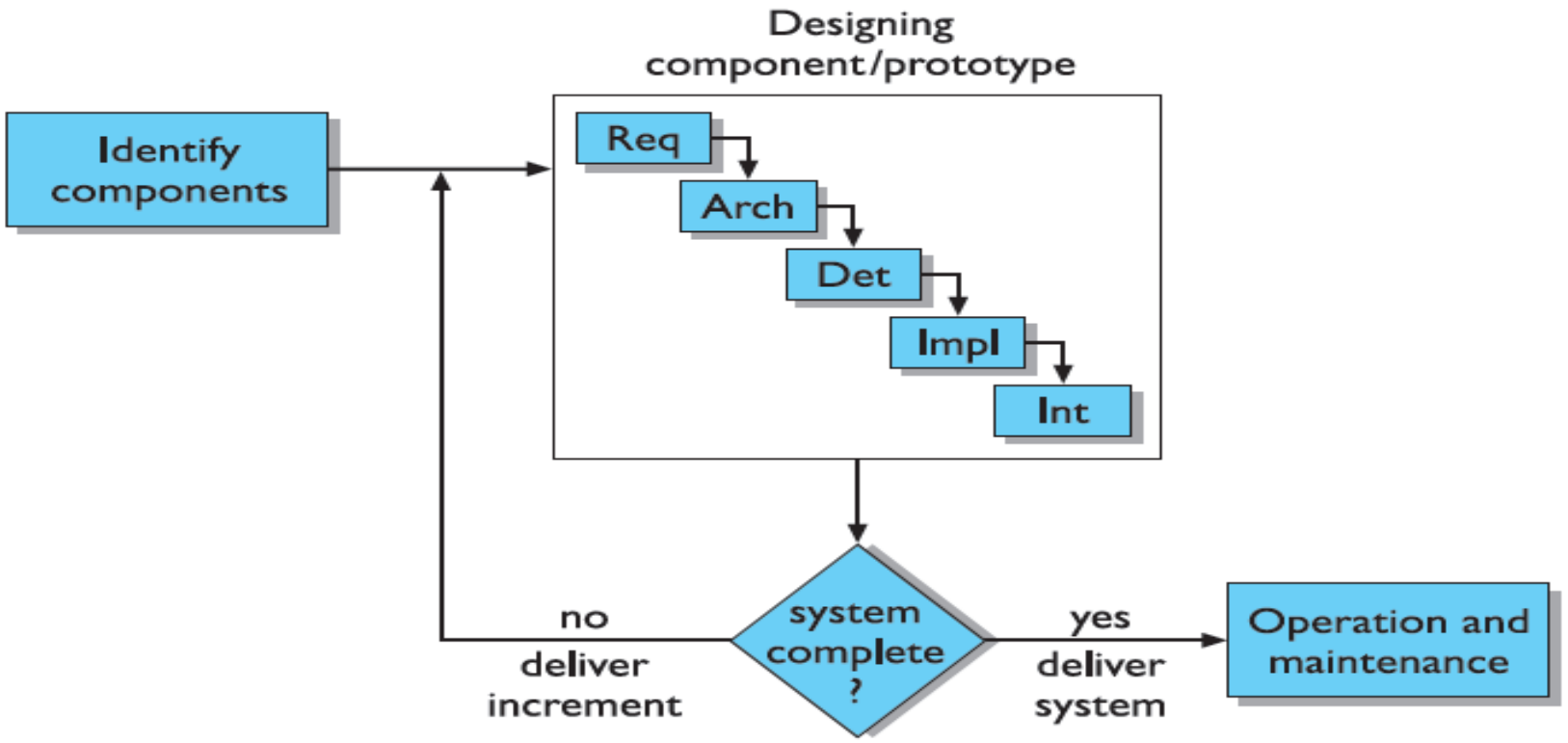
- **Iterative design and prototyping**



Throw-away prototyping within requirements specification

HCI in the Software Process

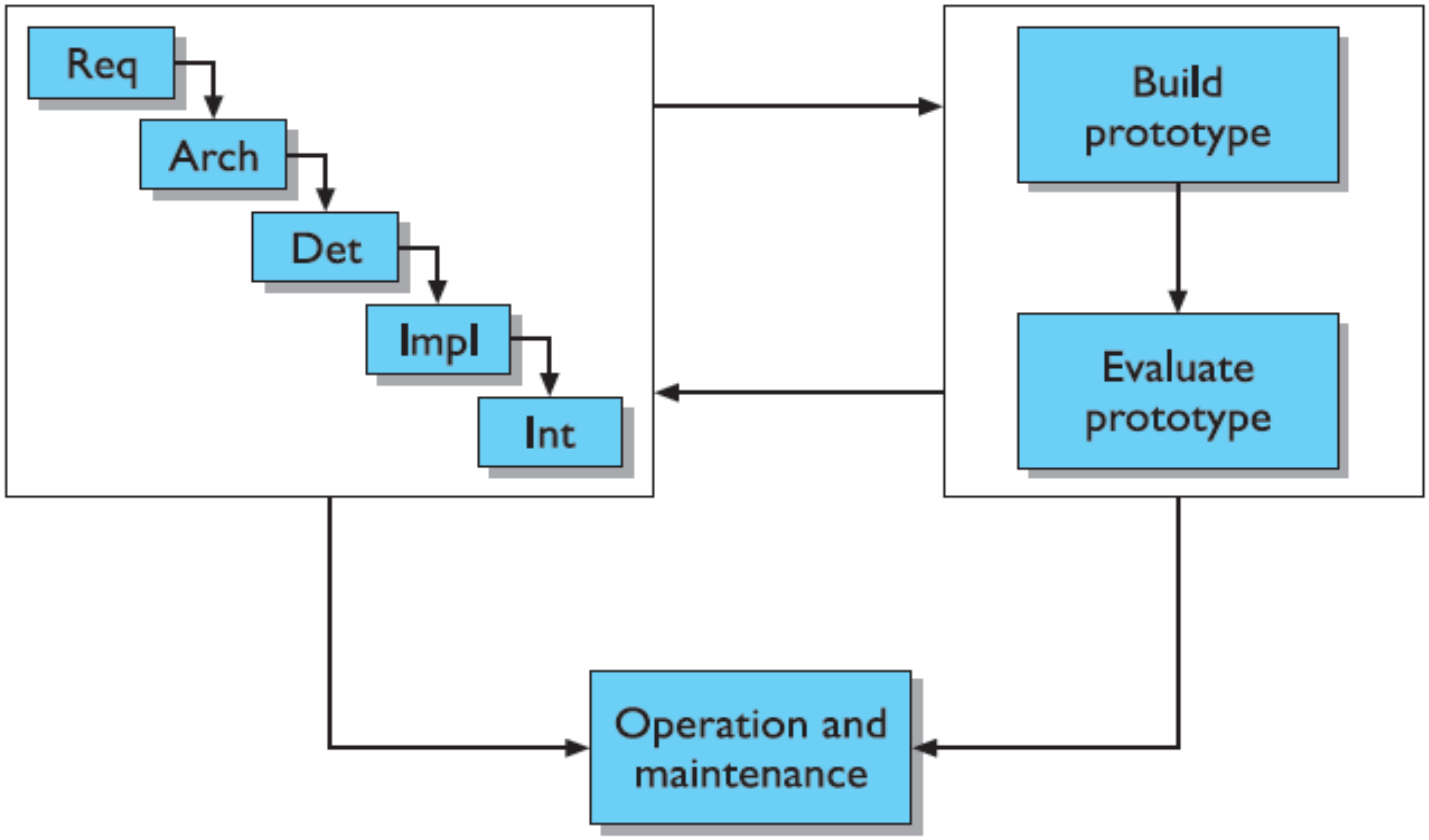
- **Iterative design and prototyping**



Incremental prototyping *within* the life cycle

HCI in the Software Process

- **Iterative design and prototyping**



Evolutionary prototyping *throughout* the life cycle

HCI in the Software Process

➢ Techniques for prototyping

- Storyboards
  - ✓ need not be computer-based
  - ✓ can be animated
- Limited functionality simulations
  - ✓ some part of system functionality provided by designers
  - ✓ tools like HyperCard are common for these
  - ✓ Wizard of Oz technique
- Warning about iterative design
  - ✓ design inertia – early bad decisions stay bad
  - ✓ diagnosing real usability problems in prototypes….
  - ✓ …. and not just the symptoms

HCI in the Software Process

➢ Design rationale

❖ Design rationale is information that explains why a computer system is the way it is.

❖Benefits of design rationale

- communication throughout life cycle
- reuse of design knowledge across products
- enforces design discipline
- presents arguments for design trade-offs
- organizes potentially large design space
- capturing contextual information

HCI in the Software Process

➢ Design rationale (cont'd)

❖ Types of DR:

- Process-oriented
  - preserves order of deliberation and decision-making
- Structure-oriented
  - emphasizes post hoc structuring of considered design alternatives
- Two examples:
  - Issue-based information system (IBIS)
  - Design space analysis

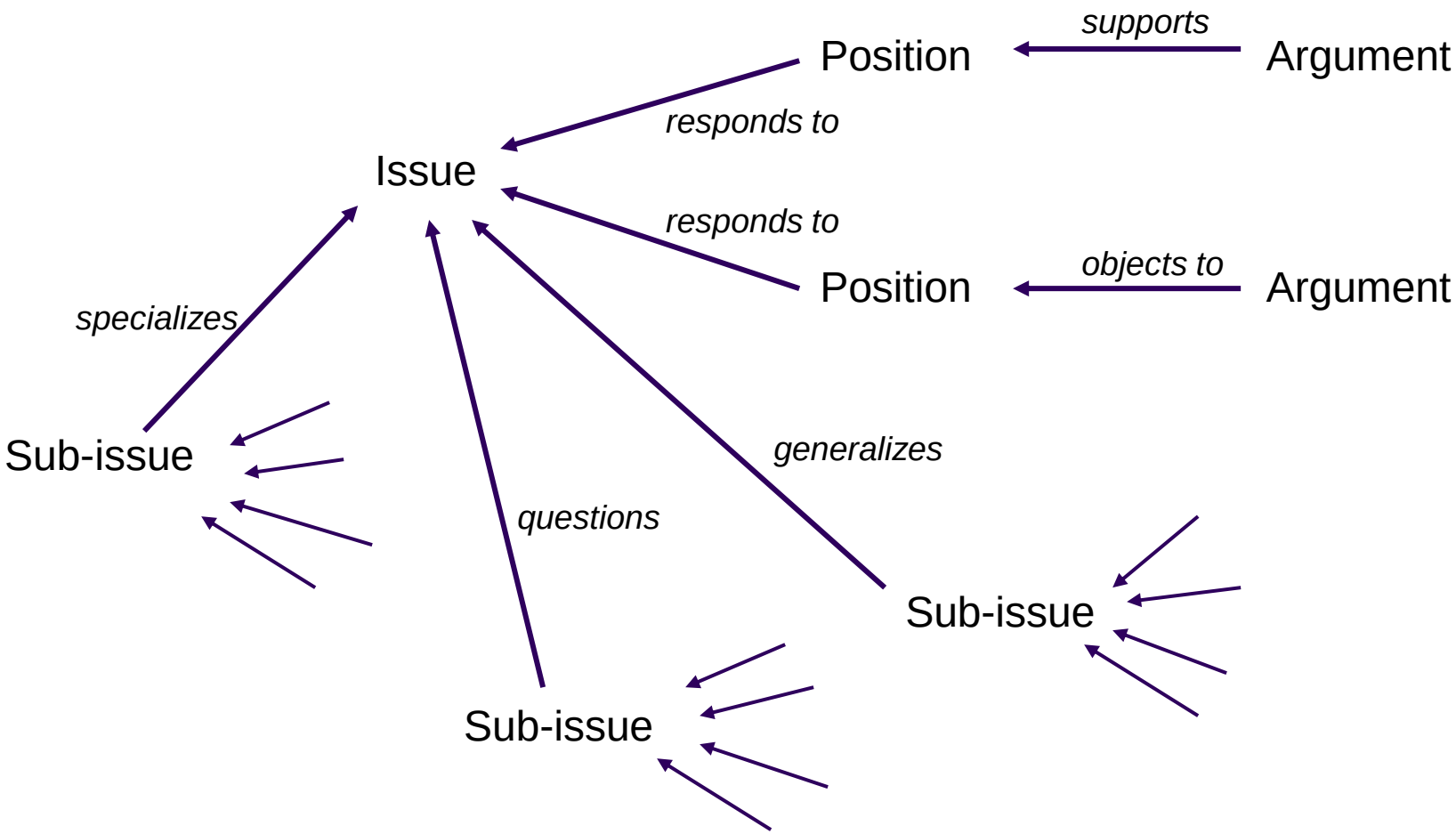
HCI in the Software Process

➢ Issue-based information system (IBIS)

- Basis for much of design rationale research
- Process-oriented
- Main elements:
  - **Issues**
    - ✓ hierarchical structure with one ‘root’ issue
  - **Positions**
    - ✓ potential resolutions of an issue
  - **Arguments**
    - ✓ modify the relationship between positions and issues
- gIBIS is a graphical version

HCI in the Software Process

➢ Structure of gIBIS



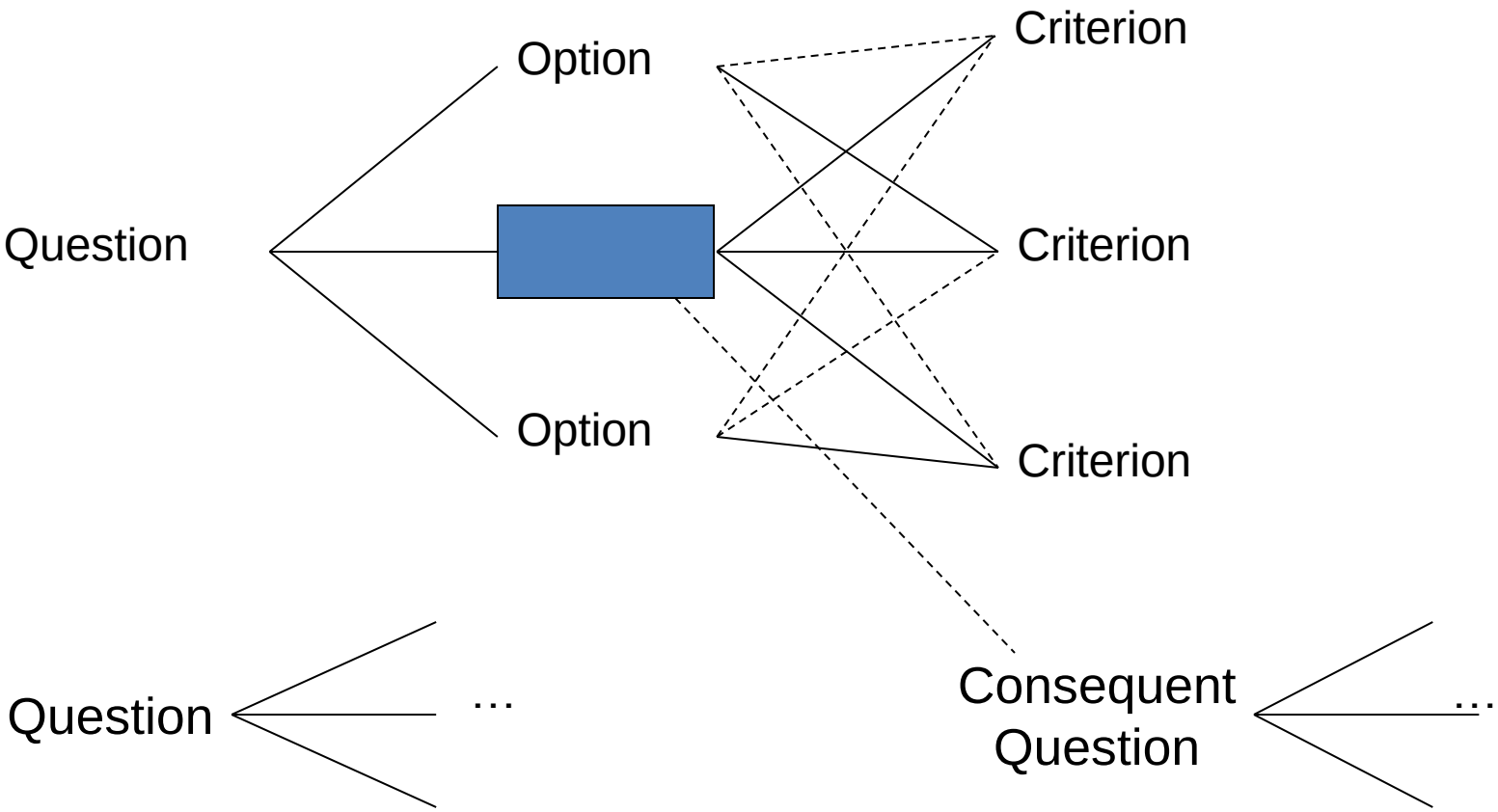
HCI in the Software Process

➢ Design space analysis

- structure-oriented
- QOC – hierarchical structure:
  - questions (and sub-questions)
    - ✓ represent major issues of a design
  - options
    - ✓ provide alternative solutions to the question
  - criteria
    - ✓ the means to assess the options in order to make a choice
- DRL – similar to QOC with a larger language and more formal semantics

HCI in the Software Process

- **the QOC notation**



HCI in the Software Process

➢ Psychological design rationale

- To support task-artefact cycle in which user tasks are affected by the systems they use
- Aims to make explicit consequences of design for users
- Designers identify tasks system will support
- Scenarios are suggested to test task
- Users are observed on system
- Psychological claims of system made explicit
- Negative aspects of design can be used to improve next iteration of design

HCI in the Software Process

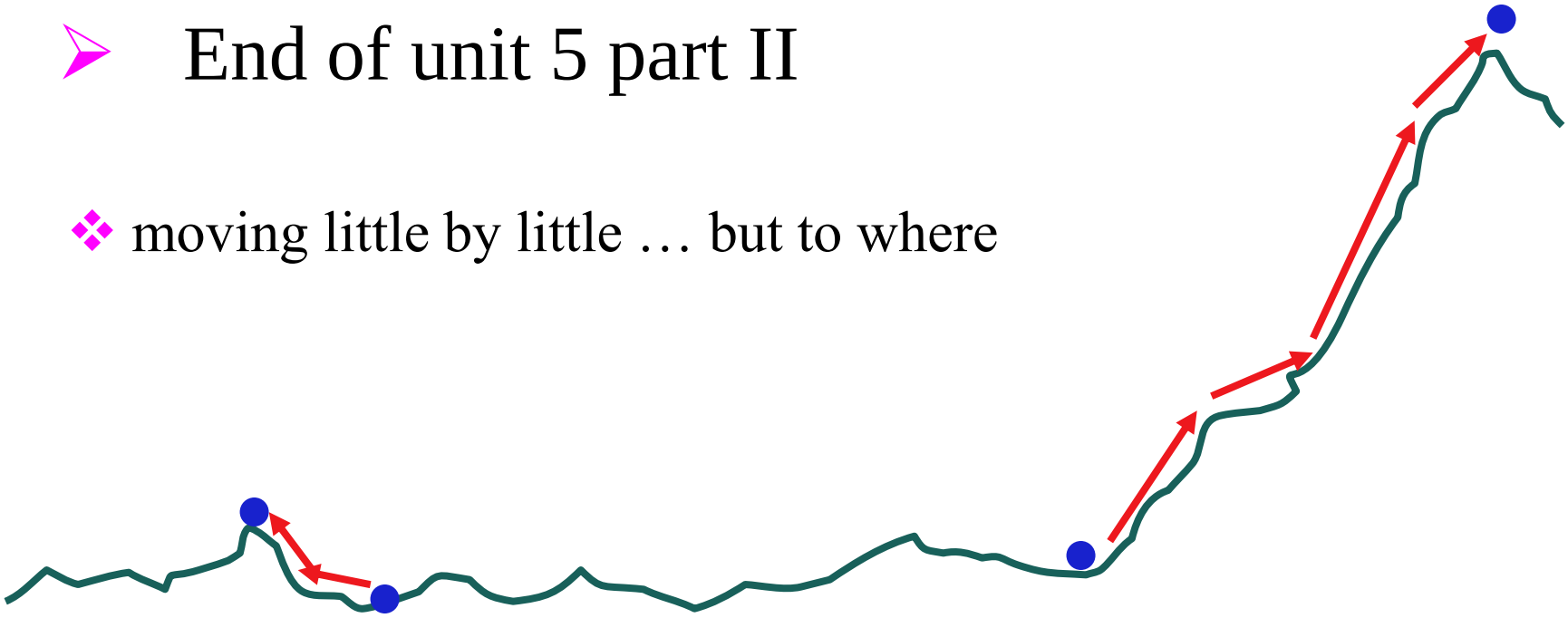
Summary

- The software engineering life cycle
  - distinct activities and the consequences for interactive system design
- Usability engineering
  - making usability measurements explicit as requirements
- Iterative design and prototyping
  - limited functionality simulations and animations
- Design rationale
  - recording design knowledge
  - process vs. structure

HCI in the Software Process

- End of unit 5 part II

  - moving little by little … but to where



1. Need a good start point
2. Need to understand what is wrong