

DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF TORONTO

CSC428F/2514F

HUMAN-COMPUTER INTERACTION

Lecture 3

INTERACTIVE SYSTEM DESIGN METHODOLOGIES
AND HUMAN-COMPUTER INTERACTION

3.1 Interactive System Design and HCI	2
3.2 The Waterfall model of software development.....	2
3.3 The Spiral model of software development.....	4
3.4 The Newman and Lamming design methodology.....	6
3.5 The BGBG design methodology	7
3.6 The Behavioural Team design methodology.....	8

Ronald Baecker
Professor of Computer Science,
Electrical and Computer Engineering, and Management
University of Toronto

Copyright © 1997, Ronald Baecker.
All rights reserved.

3.1 Interactive System Design and HCI

We cannot just be interested only in the interface

Rather we must be concerned with designing the entire system

Not just interaction, but also functionality

Not just usability, but also usefulness

Recall TimeStore

- Novel methods of representing and manipulating electronic information

- But also the functionality, including time-based information management, full-text retrieval, etc.

This brings us into the domain of software engineering

3.2 The Waterfall model of software development

Organizes development process into manageable stages

(Fig. 3.1)

- Specifies order of progression through stages

Suggests only minimal feedback loops between adjacent stages

Doesn't work for interactive systems and end-user applications

- Requirements only understood well through the development process

- Users must be consulted throughout the development process

Figure 3.1 The Waterfall Model of the Software Life Cycle
(Boehm, 1988; from BGBG, p. 282)

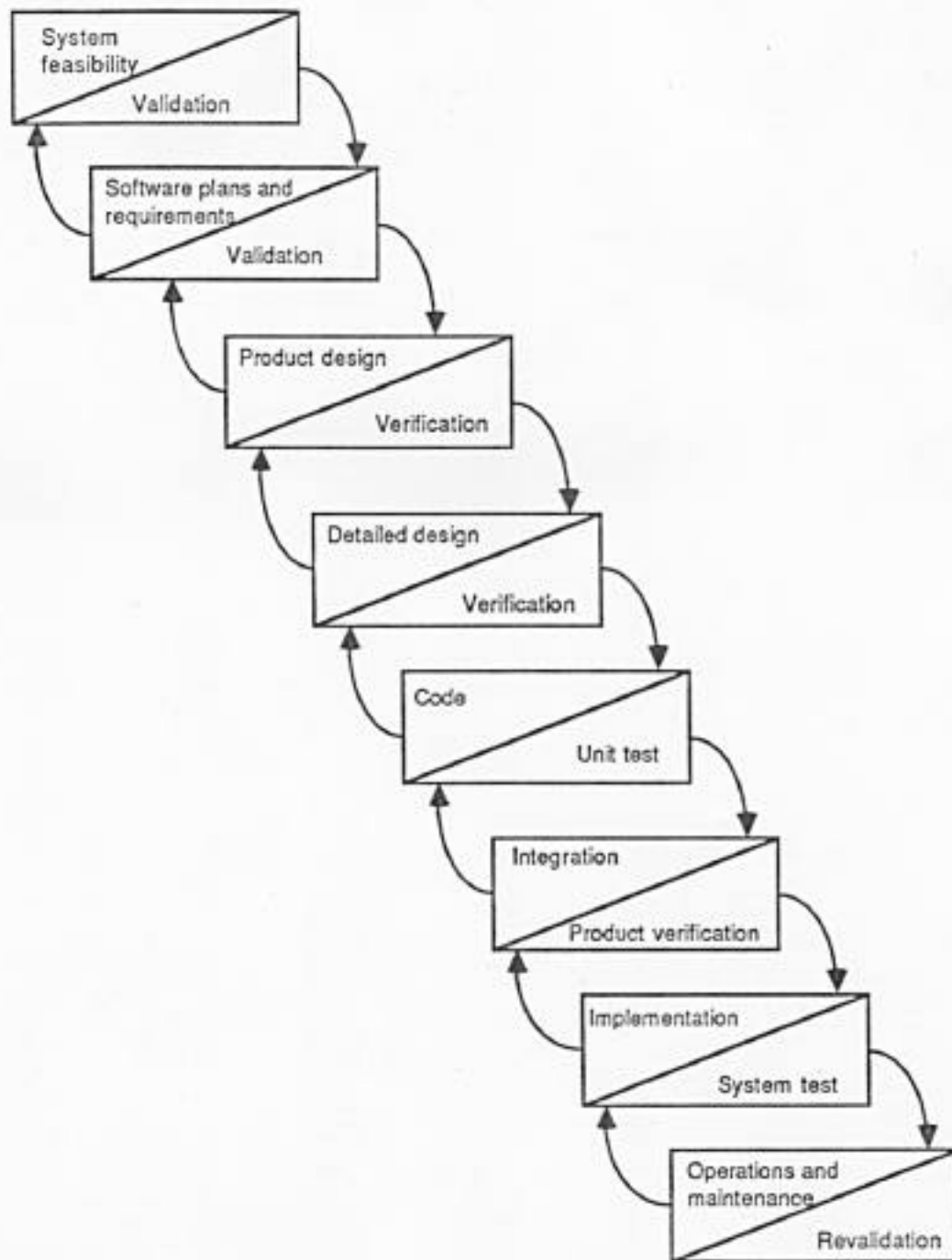


Figure 1. The waterfall model of the software life cycle.

3.3 The Spiral model of software development

One of the architects of the waterfall model, Barry Boehm, proposed this Spiral model to deal with the deficiencies of the waterfall model

Views process as successive cycles of elaboration from idea to finished system (Fig. 3.2)

Each cycle contains elements of planning, specification, prototyping, design, development, and testing

This allows for management of risk, and for refining the product concept as experience with users is gained

But spiral model is needlessly complicated, designed in part for large-scale U.S. military applications

Let's look at three other formulations of how systems can be designed for and with users

Figure 3.2 The Spiral Model of the Software Process
(Boehm, 1988; from BGBG, p. 284)

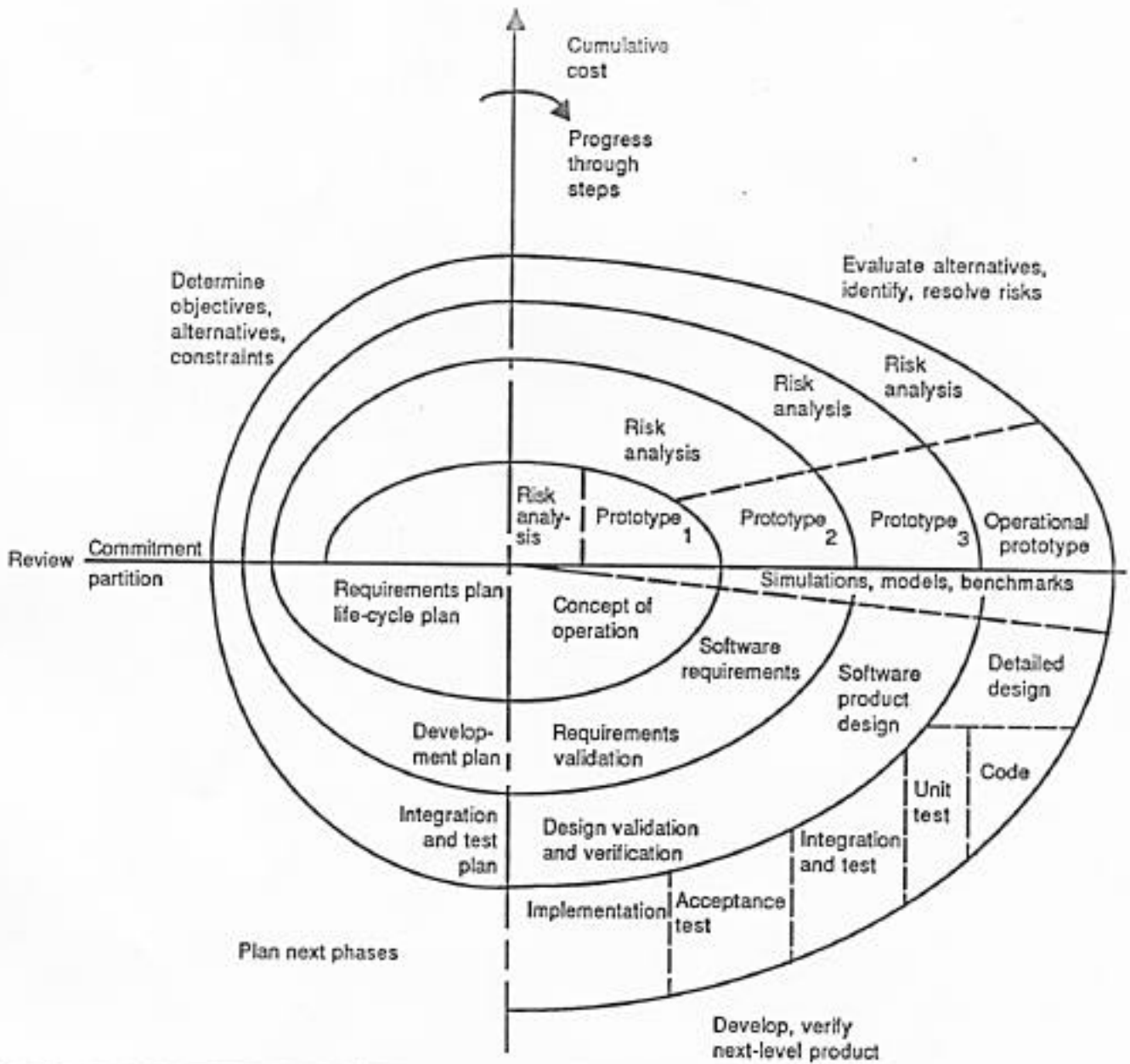


Figure 2. Spiral model of the software process.

3.4 The Newman and Lamming design methodology

Interactive system design as incorporating five kinds of concurrent processes

User study (Lectures 4-9)

Model building (Lectures 10-11)

Specification (Lectures 12-14)

Analysis of the design (Lectures 15-17)

Evaluation of prototypes (Lectures 18-21)

Diagram illustrating how these processes interconnect (Fig. 3.3)

Figure 3.3 Some of the Processes and Representations of Interactive System Design (Newman and Lamming, 1995, p. 71)

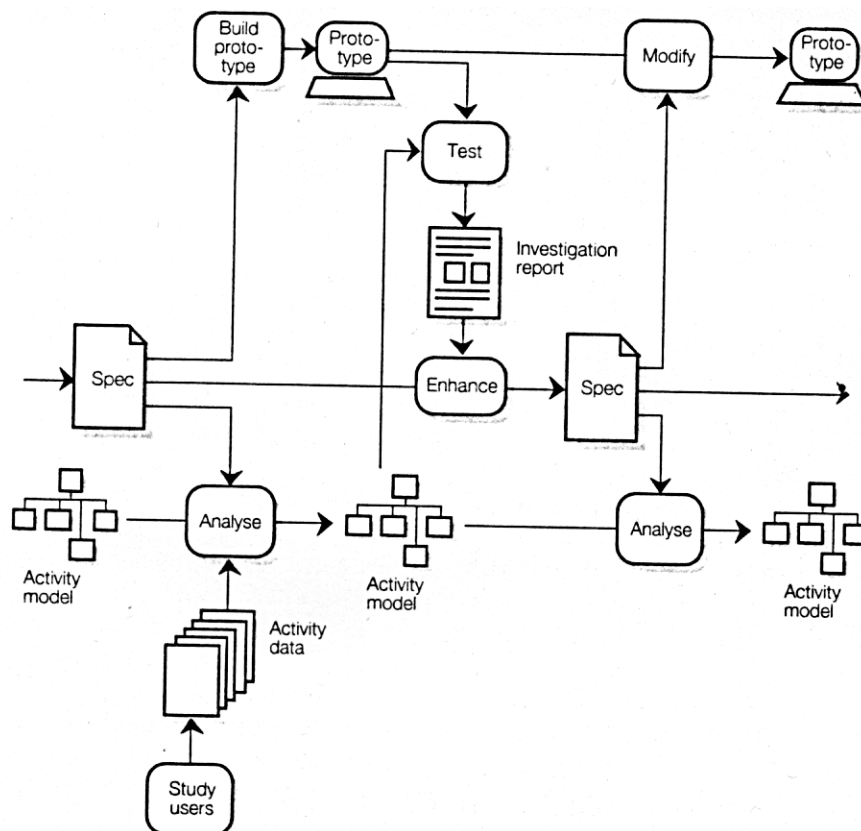


Figure 4.1 Some of the processes and representations of interactive system design.

3.5 The BGBG design methodology

Realizes iterative, user-centred design philosophy (Lecture 2)
 Not intended as a rigid formula, but as an illustration of
 a philosophy and as examples of how to proceed

Design --> Prototype --> Analyze/evaluate

--> Redesign --> Implement --> Analyze/evaluate

--> Redesign --> Revise implementation -->

Analyze/evaluate-->etc.

	Design ----->	Implement --->	Evaluate ----->
Information Collection	Questionnaires Interviews with discipline specialists Characterizations of users and tasks Days in the life scenarios		
Concept design	Initial design concepts	Design mockups	Interviews with users to get reactions to design mockups
Functionality and interface design	Design of system functionality and look-and-feel	Implementation of "smoke and mirrors" prototype	Heuristic evaluation Experiments
Prototype implementation	Design of "critical mass" of system	Implementation of partial working system	Usability tests Model-based analysis
Deliverable system implementation	Design and modification of deliverable system	Implementation and installation of this system	Usability tests
System enhancement and evolution	Design of monitoring and feedback system	Implementation of this system	Interviews and questionnaires, data collection and analysis

Information collection and evaluation (Lecture 4)

Interviews and questionnaires (Lectures 5-6)

Psychological underpinnings for good design (Lectures 7-9)

High-level system design (Lectures 10-11)

Implementation tools and methodologies (Lectures 12-14)

Model-based analysis (Lectures 15-17)

Usability testing using video (Lecture 18)

Experiments (Lectures 19-21)

3.6 The Behavioural Team design methodology — Place of Human Factors in the Process of Systems Development

1. Systems are bionic using pieces of people and pieces of machines, and let us hope, done with the right pieces
2. Roles of the usability specialist
 1. organizational fulfillment (grad students take note!)
 2. technical support
 3. advocate of user-centred thinking
3. Phases in the cycle of development
 1. Market Research: in a sense, all we do is Market Research

2. Modeling and Mock-ups: the art of supporting development
3. Lab tests: the science of evaluation
4. [No name phase]: keep in mind George Santayana's dictum that those who forget history are doomed to repeat it.

Three images (downloadable with Acrobat pdf):
Flowsuperchart
Flowchart A
Flowchart B