PART 2: THE PROCESS

AN ITERATIVE, EVALUATION-CENTERED LIFE CYCLE FOR INTERACTION DEVELOPMENT

Chapter 4

TOPICS:

• Connections of user interaction development to software engineering

• Development activities in the usability engineering life cycle

• Usability management

• User interface development team

Copyright © 2001 H. Rex Hartson and Deborah Hix.

All rights reserved. No part of this material may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, microfilming, recording, or otherwise, without prior written permission of its authors.
THE PROCESS OF USER INTERACTION DEVELOPMENT

• Connections of user interaction development to software engineering

• All these figures depict communication paths, not temporal ordering of activities

• Distinction between software design and implementation

System Development, WISCY -->

2.2 LC
THE PROCESS OF USER INTERACTION DEVELOPMENT

- Adding systems analysis, testing, and problem (application) domain
THE PROCESS OF USER INTERACTION DEVELOPMENT

• Analogous activities for user interface development

• The rest of this course is about a life cycle process for just the "User interface interaction design" box
THE PROCESS OF USER INTERACTION DEVELOPMENT

- Connecting the processes together and adding rapid prototyping

Neanderthals ->->->
LSS - USABILITY ENGINEERING LIFE CYCLE

- Iterative, evaluation-centered process model for interaction development

- New life cycle concept comes from:

* The waterfall model - movement toward completion
LSS - USABILITY ENGINEERING LIFE CYCLE

* Boehm's spiral model - evolution through iteration

Figure 2. Spiral model of the software process.
LSS - USABILITY ENGINEERING LIFE CYCLE

* Star (Hartson & Hix, 1989) - evaluation centered

* LUCID (Cognetics, Inc.) - development activities

* New work by Helms & Hartson (2001) - put it together
LSS - USABILITY ENGINEERING LIFE CYCLE

- LSS = LUCID-Star-Star = (LUCID-Star)*
- Evolution

* Spiral model abstracted
LSS - USABILITY ENGINEERING LIFE CYCLE

* Spiral unwound

![Diagram of LSS Usability Engineering Life Cycle](image)
LSS - USABILITY ENGINEERING LIFE CYCLE

* Spiral unwound and stretched out
LSS - USABILITY ENGINEERING LIFE CYCLE

* Waterfall with whirlpools
LSS - USABILITY ENGINEERING LIFE CYCLE

• Life cycle = iterative configuration of cycles (each associated with a stage/form of the product)

* Example: Systems analysis model, scenarios and screen designs, lo-fi prototype, hi-fi prototype, software production & integration

• Cycle = sequence of activities of four types:

* Analyze
* Design
* Build
* Evaluate
LSS - USABILITY ENGINEERING LIFE CYCLE

DESIGN

ANALYZE
Cycle of Activities

IMPLEMENT

EVALUATE

Intra-cycle iteration

Product In

Product

Product Out

2.15 LC
LSS - USABILITY ENGINEERING LIFE CYCLE

• Zoom in on details of a cycle activity

• Choose a technique for each activity type in each cycle
2.17 LC
LSS - USABILITY ENGINEERING LIFE CYCLE

* Shows how each activity type is assigned a technique, the role of the person doing the activity, support tools, and documentation

- Example: In scenario and screen design cycles, design walk-through is used as evaluation technique

• Basic principles

* Process is product-oriented

* Products evolve through cycles (cycles named for product forms)

* Each cycle is iterative

* Each cycle contains same activities types

* Each cycle is evaluation-centered

* Work products (documentation) evolve over cycles
LSS - USABILITY ENGINEERING LIFE CYCLE

* Process can be viewed at different levels
- Process, cycle, activity, documentation

* Any part of the process is an instance of what is possible
- Pick and choose cycles, activities, iterations to meet schedule, budget, management style

* Integrated with software engineering development process

• Validated by Helms & Hartson (2001) in e-commerce start-up company
LSS - USABILITY ENGINEERING LIFE CYCLE

• Any part of process is instance of what is possible

* Developers include their favorite UE methods and techniques

* Omit activities, cycles, iterations indicated by management, budget, schedule, project scope, team composition

* Can be different every time
USABILITY MANAGEMENT

Cycles are circles that never end...but the process ends:

• The *control* mechanism for the iteration

• Control involves:

  * Establishing usability specifications

  * Evaluating against usability specifications

    To identify problem areas

  * Performing impact and cost/benefit analyses

    To know what problems to address, in order of importance

  * Deciding on changes to make to interaction design

  * Deciding when to stop iterating

    90—10 rule: Last 10% can take 90% of time, effort

• Same process can be applied to develop user documentation or training course
USER INTERFACE DEVELOPMENT TEAM

• Roles on user interface development team

Note: Different roles, but not necessarily different people

* User interaction designer (or usability engineer or usability specialist)

* Evaluator

Some places call user the evaluator

* User (and/or user representative)

* Software engineer (or developer or analyst)

* Technical writer

* Graphic designer

* Application domain expert (also called subject-matter expert)

Expert user, content expert

Dilbert, mobius strip ->--->-