

RAPID PROTOTYPING AND ITS ROLE IN DEVELOPMENT AND EVALUATION OF USER INTERACTION

Chapter 9

TOPICS:

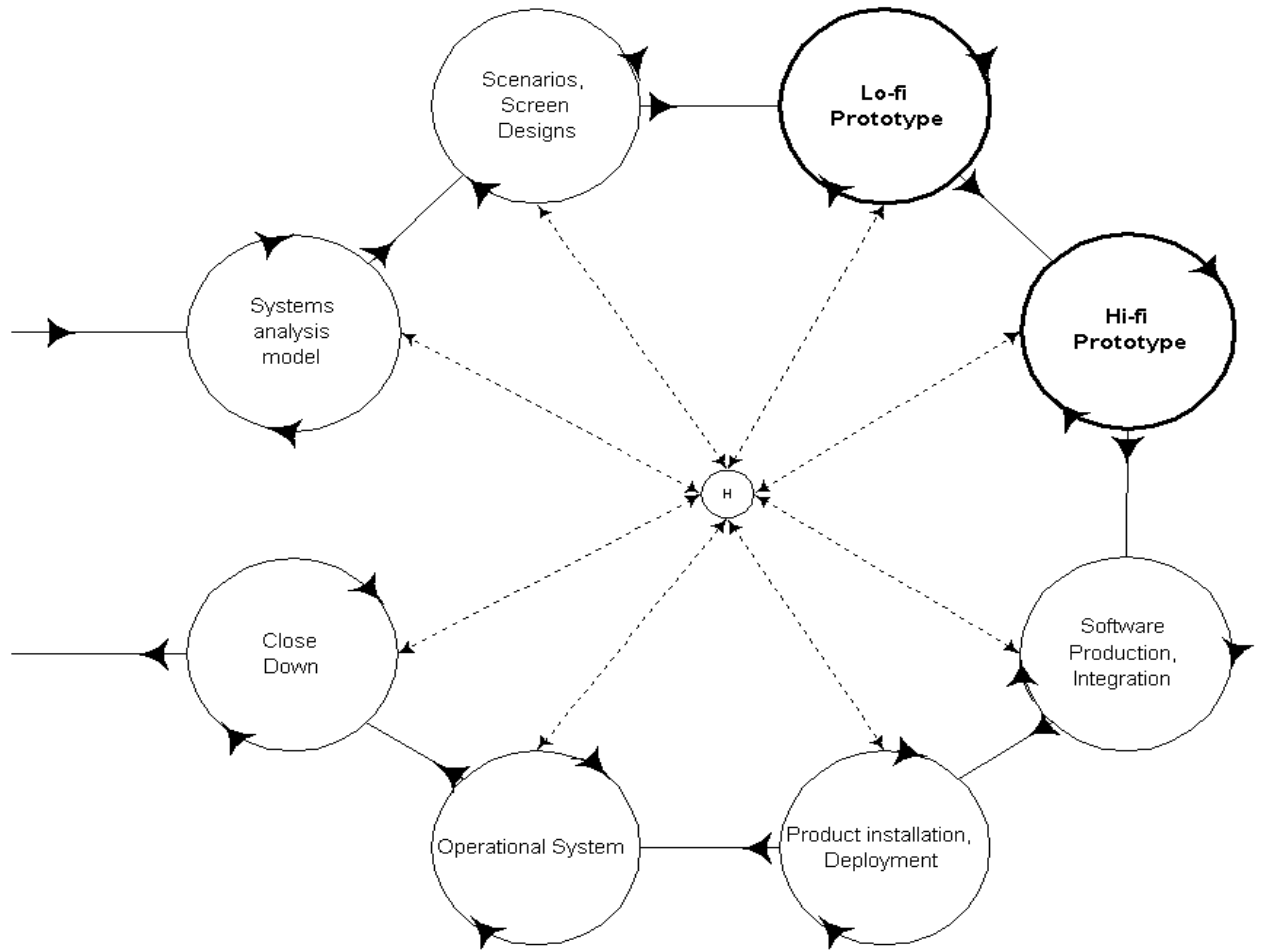
- Relation to usability engineering life cycle and iterative refinement
- Advantages and dangers of prototyping
- Low-fidelity paper prototypes
- Prototype evolution
- What to put in a prototype
- Team exercise on rapid prototyping

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.

INTRODUCTION TO RAPID PROTOTYPING

- Revisiting the usability engineering life cycle



RAPID PROTOTYPING

- Usability engineering life cycle is evaluation-centered
- Dilemma: Can't evaluate an interface until it is built, but after building, changes are difficult
- Solution: Rapid prototyping — producing interactive versions of an evolving interaction design

Don't wait until first release or field test

- * Main technique supporting iterative refinement
- Prototype is conversational "prop" to support communication of concepts not easily conveyed verbally [R. Bellamy, Apple Corp.]

ADVANTAGES OF RAPID PROTOTYPING

- Concrete baseline for communication between users and developers
- Allows user to "take it for a spin"
- Encourages early user participation and involvement
- Allows early observation of user performance
- Low fidelity prototype is obviously not finished, so users have impression it is easy to change
- Allows immediate observation of consequences of design decisions
- Can help with user "buy in"
- Can help sell management an idea for a new product
- Can help effect a paradigm shift from existing system to new system

DANGERS OF RAPID PROTOTYPING

- Needs cooperation of management, developers, and users
- Managers may view prototyping as wasteful
- Managers and/or customers and/or marketing may view prototype as final product

Dilbert "mockup" cartoon ->->->

- Programmers may lose discipline
- Prototype can be overworked (reason for prototype is forgotten)
- Prototyping tool may influence design
- Possibility of overpromising with prototype

LOW-FIDELITY PROTOTYPING

- Low-fidelity paper prototypes are bona fide technique on their own
- * Not just a low-tech substitute for computer-based prototype

Not just something you do if you don't have good software tools

- * Major corporations with extensive resources use paper prototype routinely for early interaction development
- * Computer-based prototype can distract from usability focus early on
- * People *do* take paper prototypes seriously; they do find many usability problems
- * Low-fidelity prototypes find many usability problems, and these are generally the more severe problems

HIGH- AND LOW-FIDELITY PROTOTYPING

- Interaction design has two parts
- * Look and feel: objects
- * Sequencing: behavior, including changes to object behavior

Type of prototype	"Strength"	When in life cycle to apply "strength"	Cost to fix look and feel	Cost to fix sequencing
Paper (lo-fi)	Flexibility; easy to change sequencing, overall behavior	Early	Almost none	Low
Computer (hi-fi)	Fidelity of look and feel	Later	Low	High

PROTOTYPE EVOLUTION WITHIN A PROJECT

"Product" type	Purpose (in development)
Scenarios and screen designs	Requirements gathering, client/user walk-throughs
Hand-drawn paper prototype	Evaluate conceptual model, early design ideas
Computer-printed paper prototype (e.g., with VB, Visio)	Early formative usability evaluation
Computer-based prototype with some working functionality (e.g., database functions)	Primary formative usability evaluation
Product release	Field support, customer help line, post-deployment usability evaluation

WHAT TO PUT IN A PROTOTYPE

- What to put in early prototypes — to evaluate usability of overall interaction design metaphor/paradigm (conceptual design)

- * Low fidelity

- * Start with *representative* sample screen or two

- * Mock-up a *representative* task

- * Follow a *representative* task thread

Learn a great deal from incomplete design, and from a single brand new user

- What to put in later prototypes — to evaluate usability of details (detailed design)

- * High-fidelity

- * More detailed, refined screens

- * More complete tasks

Dilbert: feature creep ->->->

WHAT TO PUT IN A PROTOTYPE

- Same process to prototype Web applications as for GUIs

Web can make prototype more accessible for different users, conditions

- Invest *just enough* effort in a rapid prototype
- * To achieve formative evaluation goal, but no more
- Typical transition to real product
- * Keep details (code) of user interaction design objects, look and feel
- * Discard any functional or other code holding prototype together

E.g., sequencing code; was never meant to be product code

Dilbert: Dewdrop, beta

TEAM EXERCISE: RAPID PROTOTYPING

ALMOST ALL YOU EVER WANTED TO KNOW ABOUT RP YOU LEARNED IN KINDERGARTEN!!!

- *Goal:*
 - * To obtain experience with rapid construction of a low fidelity prototype for early stages of user interaction design
- *Activities:*
 - * Draw Web pages in more detail than in your scenarios
 - * Make prototype "executable," at least for benchmark tasks
- *General – What we are going to do:*
 - * Draw interaction objects on paper, cut them out, and tape in aligned position, relative to other objects, on separate blank plastic transparencies.
 - * Use "easel" to register each sheet of plastic with other sheets.

TEAM EXERCISE: RAPID PROTOTYPING

- * During "execution" most dynamics will be created by adding and removing various registered plastic sheets to/from the easel.
- * You will need to prototype at least the benchmark tasks from your usability specifications, since the prototype will be used in the formative evaluation exercise. Prototype will be "executed" on the easel, usually taped to tabletop for stability.
- * **IMPORTANT:** Get *everyone* on your team involved in drawing, cutting, taping, etc. — not just one or two people. You'll be done much faster if everyone pitches in. However, this is not art class, so don't worry too much about straight lines, exact details, etc.

TEAM EXERCISE: RAPID PROTOTYPING

- * Start with simplest possible background for each Web page **in pencil or pen on full size paper**, as base for all moving parts
 - Include only parts that never change (e.g. for Y2K Calendar: monthly "grid", no month name)
- * Everything else is **drawn in pencil or pen on paper**, cut out, and taped (in proper location) on separate plastic sheet
- * Don't draw *anything* twice; make it modular to reuse
 - The less you put on each layer, the more modular
- * Whatever changes when user gives input should go on *separate* paper-on-plastic sheet

TEAM EXERCISE: RAPID PROTOTYPING

- * If user will **type in values** (e.g., item number) use clear sheet on top and marking pen
- * Make a **highlight** for major selectable objects
 - Use square or rectangle with "handle"; color with marking pen
- * Fasten some objects (e.g., pull-down lists) to top or side of easel with tape "hinges", so they "flap down" to overlay the screen
- * Use any creative techniques to demonstrate motion, dynamics, feedback
 - E.g., scrolling can be done with paper through slits cut in larger paper (all taped to plastic sheet)
- * Make a "not yet implemented" (or "under construction") message!

TEAM EXERCISE: RAPID PROTOTYPING

* *Pilot testing*: Be sure that your prototype will support your benchmark tasks by having one member of your team "run" the prototype while another member plays "user" and tries out the benchmark tasks.

- *Deliverables*:

>>>> An "executable" version of your prototype, constructed of paper taped in registration to plastic sheets.

>>>> Pilot test completed

- *Completed by*:

2 hours max.