

Design Basics

CS5540 HCI

Rich Riesenfeld
Fall 2004

Thesis

- HCI intrinsically involves *design*
 - “*Design an interface to ...*”
- What does this observation entail?

Whereas...

- Design is as old as creativity
- Intensively studied subject
- Much is known
- Let's tap this understanding and experience!

Design is Ubiquitous

- Nearly *all* human activities involve design
 - Novels, airplanes, murals...
 - Rescue missions, ascents...
 - Algorithms, software, interfaces

Design Approaches

- Top down
 - Mechanical linkages, compilers, software system
 - Airplane, eg: mission, configuration, weight
 - Recursive refinement technique
- Particular as an instance of General
 - Parametric design

Fall 2003

Utah School of Computing

Student Name Server

slide 5

Design Approaches (2)

- Bottom up
 - Prototype, gain experience
 - Abstract principles
 - Scale up; begin slow
- Infer *General* from *Particular*
 - *Linguistics*

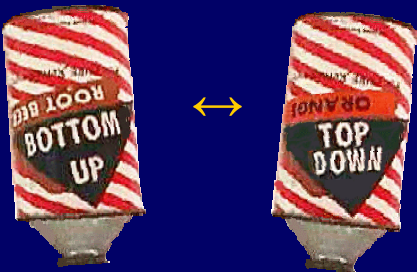
Fall 2003

Utah School of Computing

Student Name Server

slide 6

Note: Bottom Up ↔ Top Down



Fall 2003

Utah School of Computing

Student Name Server

slide 7

Design Challenges

- Economics
 - Make it good and cheap
 - "*Better, faster, cheaper*"
- Constraints
 - Not design without constraints

Fall 2003

Utah School of Computing

Student Name Server

slide 8

Critical Choices

- Design involves making wise “trade-off”
 - Form v. function
 - Weight v. durability
 - Specific and focused v. general and diffuse
 - *Paint v PhotoShop*
 - Etc. ...

Fall 2003

Utah School of Computing

Student Name Server

slide 9

Design Integrity

- Clear purpose
 - Understand the role
 - Who is user and what is her profile?
 - Good functional spec
 - Tasks to accomplish?
 - Who is user?
 - Budgets?

Fall 2003

Utah School of Computing

Student Name Server

slide 10

Design Discipline

- Maintain focus and charge
 - Refer to specs often
- Creeping “feature-ism”
 - “Wheel of re-incarnation” (IES)
 - Compact cars, portable models, basic models, etc.
 - Features are NOT free!

Fall 2003

Utah School of Computing

Student Name Server

slide 11

Design Discipline (2)

- Sunset the lifecycle
- Expanded *spec*
- *New technologies change “design equations”*
- “*Just shoot it*”
 - *Start over!*

Fall 2003

Utah School of Computing

Student Name Server

slide 12

Design Phases/Stages

- Conceptual
 - Show that idea can work
- Preliminary
 - Sufficient to understand, cost, etc
- Detail
 - The "whole enchilada"
 - Adequate for contracting

Fall 2003

Utah School of Computing

Student Name Server

slide 13

"Design Intent"

- Why did the designer do this?
- What is the function of this component?
- What was the designer thinking?
- What are the implications if this is modified?

Fall 2003

Utah School of Computing

Student Name Server

slide 14

Design History

- Better at design than documentation
- Not sensitive to capturing the past
- Important for the future of a product
- Need better tools
- Record the history as well as final result!

Fall 2003

Utah School of Computing

Student Name Server

slide 15

Documentation

- Should not be a post-process
- Capture at time of creation
- Hard problem, actually
 - Who should do it?
 - How should it be accomplished?
- Expensive
 - Not always part of deliverable!

Fall 2003

Utah School of Computing

Student Name Server

slide 16

Design Conventions

- Use standards for components
- Use standards for style
- Don't re-invent terms, tech, tools, etc.
- Make it as straightforward as possible for others who work with you

Fall 2003

Utah School of Computing

Student Name Server

slide 17

Variant Design

- Most designs are not really new from the bottom up!
- *Redesign* is far more common as an activity than design, actually
- Make use of the past
- Use templates, components, previous knowledge, catalogs, etc.

Fall 2003

Utah School of Computing

Student Name Server

slide 18

Lifecycle Design

- Consider the entire life of a product
 - Cradle to grave (incl disposal)
 - Look at lifecycle cost!
 - Who will maintain?
 - How long will product live?
 - What tools are appropriate?
 - Situations change!

Fall 2003

Utah School of Computing

Student Name Server

slide 19

Design for Change

- The only sure thing about a design is that its requirements will evolve and may change dramatically
- Build it flexibly, modularly, clearly wrt to intent, etc

Fall 2003

Utah School of Computing

Student Name Server

slide 20

Design Spiral

- Iterate repeatedly
- Budget for interaction
- Throw away early attempts as learning exercises
 - Steve Coons "I know what to throw out."

Fall 2003

Utah School of Computing

Student Name Server

slide 21

"ilities" of Design

- Maintainability
- Portability
- Readability
- Flexibility
- Testability
- Etc, etc....

Fall 2003

Utah School of Computing

Student Name Server

slide 22

Complexity "Banana"

- Complexity space often is shaped like a banana:
 - Many simple instances
 - Few complicated instances

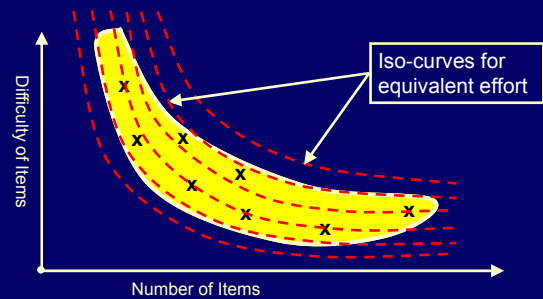
Fall 2003

Utah School of Computing

Student Name Server

slide 23

Banana Envelope



Fall 2003

Utah School of Computing

Student Name Server

slide 24

Design "Reuse"

- Try to make the parts re-usable for other things or future renovations
- Use existing parts if available and of adequate quality

Fall 2003

Utah School of Computing

Student Name Server

slide 25

Design is "team sport"

- Most designs involve more than one
- Interfaces are critical, not just components
- Communications, small granularity exchanges, important
- Negotiation, compromise part of deal

Fall 2003

Utah School of Computing

Student Name Server

slide 26

Design Views

- Components may serve different functions
 - Different designers see different views
 - Pockets v. Ribs
 - Manufacture v. Structures

Fall 2003

Utah School of Computing

Student Name Server

slide 27

Testing and Validation

- Important stuff!
- Expensive phase
- Underdone activity
 - Alpha testing
 - Beta testing

Fall 2003

Utah School of Computing

Student Name Server

slide 28

Design Review

- Take stock of progress periodically
- Is design on track?
- Have it critiqued by a group

Fall 2003

Utah School of Computing

Student Name Server

slide 29

Design Evaluation

- How well does design *perform*?
 - Consider all aspects and costs
 - Were the trade-offs wise?

Fall 2003

Utah School of Computing

Student Name Server

slide 30

Debugging Discipline

- Early is better: easier and cheaper
- Product recall is the ultimate “debugging,” and the most expensive, incl product liability

Fall 2003

Utah School of Computing

Student Name Server

slide 31

Design Safety

- Consider failure modes
- What are the consequences of failure?
- Have they been adequately explored and mitigated?

Fall 2003

Utah School of Computing

Student Name Server

slide 32

Design is a Creative Process

- Respect its needs
 - Time and patience
 - Concentration, protracted focus
 - Freedom to explore new ways
 - Liberation from past
 - Individual encouragement and support
 - Most ideas are not “keepers”

Fall 2003

Utah School of Computing

Student Name Server

slide 33

Consider Multiple Solutions

- Competing prototypes
 - Learn more about merits and liabilities
- Gain experience
- “American way (free market)...”
 - Can help evoke “best effort”

Fall 2003

Utah School of Computing

Student Name Server

slide 34

Msg: Recognize Design Activity

- Encourage good design practice
- Nurture good design through better understanding of its nature
 - Establish and protect a conducive environment
- You *are* designers! Do it well!

Fall 2003

Utah School of Computing

Student Name Server

slide 35

Utah School of
Computing

The End

Design Methodology