ICS 463
Human Computer Interaction

8. Refinement

Dan Suthers
Spring 2004

Conceptual Design

• Transform user requirements/needs into a conceptual model
  - "a description of the proposed system in terms of a set of integrated ideas and concepts about what it should do, behave and look like, that will be understandable by the users in the manner intended"

• Two fundamental approaches:
  - Creative exploration
  - Refinement

Preece's Questions

• Which interaction mode? (How the user invokes actions)
  - Activity-based: instructing, conversing, manipulating and navigating, exploring and browsing
  - Object-based: structured around real-world objects

• Which interaction paradigm?
• Is there a suitable metaphor?
**Approaches**

- We’ve got different approaches here:
  - Textbook advocates a visually oriented scenario-based design, with early consideration and prototypes
  - Rosson & Carroll’s scenario-based design is text (story) oriented version
  - Constantine & Lockwood’s Usage-Centered Design is top down refinement of models
- There’s no one right answer…
- …but there are wrong answers!
- Reflect on how the methods you choose meet the needs of your project

**Plan**

- In my graduate design course I take 4 weeks to cover the material of this chapter
- I’m reorganizing
  - Refinement from conceptual to physical design this week
  - Prototyping and involving users next week
- I’ll focus on UCD with a little SBD, and let the textbook speak for itself.

**Scenario-Based Design**

*Mostly Rosson & Carroll’s version*
Scenario-Based Design

We are now here

Scenario Transformation

- **Problem scenario:**
  - describes current problem situation
- **Activity scenario:**
  - moves to design world & envisionment, but still just emphasizing the activity
- **Information scenario:**
  - adds just the information/presentation details, along with user’s anticipation, reaction to what he/she sees
- **Interaction scenario:**
  - all the UI details and user experience, i.e. the full "story"

Activity Design
**Doing Activity Design**

- Write stories envisioning use as we’d like it to be, but without implementation details
- Work on Activity Scenario and Claims Analysis Together
  - Scenarios are a flexible and inexpensive medium to try out ideas
  - Claims analysis becomes design rationale
  - Maintain or enhance as many positive impacts as you can
  - While removing or minimizing negative impacts

**Information/Interaction Design**

**The Gulfs of HCI**

Information and Interaction correspond to the two gulfs
Information Design

Specify representations of task objects and actions that help users
- perceive,
- interpret and
- make sense of what is happening

- Elaborate the activity design scenarios with presentation details and stakeholder reactions
- Expand claims analysis to include features of the information design

Devices and their tradeoffs

- Redundant coding: use of multiple cues to code the same information
  - Increases speed and accuracy of processing
  - May reduce available dimensions or increase complexity
- White space
  - Use to separate unrelated elements
  - Excessive use means dead space
- Borders and boxes
  - Use to group related elements
  - Excessive use increases clutter
- Grid based design improves organization

Tradeoffs in Design for Perception

- Task-relevant information versus complexity
  - decompose tasks, link to less critical information
- Offer visual distinctions, but not too many levels
  - too many variations (e.g., different colors) will make the cues hard to discriminate, slowing perception
- Elegant designs exploit position, thematic repetition, low-key color schemes, and white space, instead of lines, boxes, and labels
Consistency

- **Internal consistency within a system**
  - on the same screen: button shape, fonts, etc.
  - from screen to screen: UI controls, layout, font family
  - applies to text vocabulary too
- **External consistency across different systems**
  - e.g., the Mac family of apps, Windows, the Web
  - enables transfer of learning from one system to another
  - mismatches lead to interference
- But consider special needs of user's task

Dynamic displays

- Take advantage of computational medium
- Allows us to reduce complexity by hiding information but creates need to know how to uncover that info and how to navigate
  - Focus-context
  - Other multiple coordinated views
  - Smooth panning and zooming
  - Semantic filtering:

Interaction Design

- Bridging the Gulf of Execution
- Design of the mechanisms for accessing and manipulating task information
  - what system goals are possible,
  - plans for accomplishing them,
  - physical actions to execute
- Tightly integrated with Information Design
Methods for interaction design

- Elaborate the Scenarios and Claims
- Consider implications of Metaphors and Technologies
- Storyboarding: graphical scenario showing major events

Mapping Task to System Goals

- Direct manipulation of objects analogical to real world
  - simplifies mapping from task to system goals (recognition),
  - but clutter workspace
- Command languages
  - flexible and don’t clutter workspace,
  - but are hard to learn and remember (recall)
- Compromises include
  - Commands that look like objects (WIMP)
  - Hierarchical command menus
  - Dynamic displays that hide manipulable objects

Planning Action Sequences

- Major issue is handling complexity
  - Chunk sequences that correspond to subgoals
  - Use consistent sequences
  - Forms can help guide action sequences
- Allow for multi-tasking (“multithreading”)
  - Avoid modal interaction
  - Make state visible
  - Multiple or tiled windows
Designing for Errors

Mistakes versus slips may require different handling
Some methods to consider:
- Representing available actions
- Feedback of consequences
- Confirmation dialogs (how to avoid habitual confirmation?)
- Undo (what granularity? how far?)

Usage-Centered Design

My adaptation of Constantine and Lockwood

Usage Centered Design Overview

Content model (interface architecture) models both Information and Interaction, but abstractly.
Levels of design Revisited

- **Conceptual (Activity)**
  - Task
  - Semantic
- **Physical (Implementation)**
  - Interaction (Operational)
  - Syntactic (Representational)

Interface Architecture

- **Contexts** say what content and tools go together to support some tasks
- **Navigation Map** says how you move between contexts
- Together they describe the content and organization of the interface while abstracting away from details of appearance and behavior

Mapping Task to Content Model

Oversimplifying: to translate a use case into a context, nouns become contents and verbs become tools
Not in their original formulation, this was added to bridge the gap from interface architecture to visual design.
Putting it together

Introduce the models in this order (but iterate):
1. Role models
2. Essential use cases
3. Contexts
4. Navigation Map
5. Abstract Layout
6. Pencil drawings
7. Web-based mockup or prototype

NetLearn Example

Key (Old System)

- Content display area (sometimes editable).
- Linkage display: relates content display elements
- Command tool that carries out actions within the present context.
- Command tool that takes you to another context.

Sticky notes
- Are easy to reorganize
- Are never mistaken for the real interface!
Let's build a context for this ...

Group Workspace Entry

Summary List of Assignments
Assignment Selector

News

Submit Work

Learner or Coach

Summary List of Work
Work Selector

Assignment Description

Work Upload Tool

Loader or Coach

System Responsibility

Enter study group
Direct assignment
Submit work to upload
Indicate completion
Upload
Indicate success status
Note: examples may be charts, charts, or math problems
Annotation Use Cases

Review Work

- Select work
- Display work details
- Discuss work
- Display work

Review Annotations

- Display work
- Display comments on file
- Request comments on file
- Display work details

"Detail Work" (Annotation) Display

We thought we might need to switch between examples of work ...

"Detail Work" (Annotation) Display
Let's try it ...

Continuing with the ICS Web Site from last week

- Write an Activity Scenario
  - a story of use that does not make commitments to implementation
- Fix up your Essential Use Cases to match
- Map EUC to Content Model (contexts)
- Refine Content Model to Abstract Layouts

(NAVIGATION MAP)

(will discuss next week)
Assignments

Assignment 6
- Conduct the data gathering you planned in Assignment 5
- Perform a requirements analysis

Assignment 7 (over Spring Break)
- Refinement of requirements to information/interaction design (content, abstract layout)