Chapter 8
Design, prototyping and construction
Overview

• Prototyping and construction
• Conceptual design
• Physical design
• Generating prototypes
• Support for design
Overview

• Prototyping and construction

• Conceptual design

• Physical design

• Generating prototypes

• Support for design
Prototyping and construction

- What is a prototype?
- Why prototype?
- Different kinds of prototyping
  - low fidelity
  - high fidelity
- Compromises in prototyping
  - vertical
  - horizontal
- Construction
What is a prototype?

In other design fields a prototype is a small-scale model:

• a miniature car
• a miniature building or town
• the example here comes from a 3D printer
What is a prototype?

In interaction design it can be (among other things):

- a series of screen sketches
- a storyboard, i.e. a cartoon-like series of scenes
- a Powerpoint slide show
- a video simulating the use of a system
- a lump of wood (e.g. PalmPilot)
- a cardboard mock-up
- a piece of software with limited functionality written in the target language or in another language
Why prototype?

- Evaluation and feedback are central to interaction design
- Stakeholders can see, hold, interact with a prototype more easily than a document or a drawing
- Team members can communicate effectively
- You can test out ideas for yourself
- It encourages reflection: very important aspect of design
- Prototypes answer questions, and support designers in choosing between alternatives
# Filtering dimensions of prototyping

<table>
<thead>
<tr>
<th>Filtering dimension</th>
<th>Example variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appearance</td>
<td>size; color; shape; margin; form; weight; texture; proportion; hardness; transparency; gradation; haptic; sound</td>
</tr>
<tr>
<td>Data</td>
<td>data size; data type (e.g., number; string; media); data use; privacy type; hierarchy; organization</td>
</tr>
<tr>
<td>Functionality</td>
<td>system function; users’ functionality need</td>
</tr>
<tr>
<td>Interactivity</td>
<td>input behavior; output behavior; feedback behavior; information behavior</td>
</tr>
<tr>
<td>Spatial structure</td>
<td>arrangement of interface or information elements; relationship among interface or information elements – which can be either two-or three-dimensional, intangible or tangible, or mixed</td>
</tr>
</tbody>
</table>
## Manifestation dimensions of prototyping

<table>
<thead>
<tr>
<th>Manifestation dimension</th>
<th>Definition</th>
<th>Example variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Material</td>
<td>Medium (either visible or invisible) used to form a prototype</td>
<td>Physical media, e.g., paper, wood, and plastic; tools for manipulating physical matters, e.g., knife, scissors, pen, and sandpaper; computational prototyping tools, e.g., Macromedia Flash and Visual Basic; physical computing tools, e.g., Phidgets and Basic Stamps; available existing artifacts, e.g., a beeper to simulate a heart attack</td>
</tr>
<tr>
<td>Resolution</td>
<td>Level of detail or sophistication of what is manifested (corresponding to fidelity)</td>
<td>Accuracy of performance, e.g., feedback time responding to an input by a user (giving user feedback in a paper prototype is slower than in a computer-based one); appearance details; interactivity details; realistic versus faked data</td>
</tr>
<tr>
<td>Scope</td>
<td>Range of what is covered to be manifested</td>
<td>Level of contextualization, e.g., website color scheme testing with only color scheme charts or color schemes placed in a website layout structure; book search navigation usability testing with only the book search related interface or the whole navigation interface</td>
</tr>
</tbody>
</table>
What to prototype?

- Technical issues
- Work flow, task design
- Screen layouts and information display
- Difficult, controversial, critical areas
Low-fidelity Prototyping

• Uses a medium which is unlike the final medium, e.g. paper, cardboard

• Is quick, cheap and easily changed

• Examples:
  sketches of screens, task sequences, etc
  ‘Post-it’ notes
  storyboards
  ‘Wizard-of-Oz’
Storyboards

• Often used with scenarios, bringing more detail, and a chance to role play

• It is a series of sketches showing how a user might progress through a task using the device

• Used early in design
Sketching

- Sketching is important to low-fidelity prototyping
- Don’t be inhibited about drawing ability. Practice simple symbols
Card-based prototypes

- Index cards (3 X 5 inches)
- Each card represents one screen or part of screen
- Often used in website development
‘Wizard-of-Oz’ prototyping

• The user thinks they are interacting with a computer, but a developer is responding to output rather than the system.
• Usually done early in design to understand users’ expectations
• What is ‘wrong’ with this approach?

User

> Blurb blurb
> Do this
> Why?

Design, prototyping and construction
High-fidelity prototyping

• Uses materials that you would expect to be in the final product.

• Prototype looks more like the final system than a low-fidelity version.

• For a high-fidelity software prototype common environments include Macromedia Director, Visual Basic, and Smalltalk.

• Danger that users think they have a full system.......see compromises
Compromises in prototyping

• All prototypes involve compromises
• For software-based prototyping maybe there is a slow response? sketchy icons? limited functionality?
• Two common types of compromise
  • ‘horizontal’: provide a wide range of functions, but with little detail
  • ‘vertical’: provide a lot of detail for only a few functions

• Compromises in prototypes mustn’t be ignored. Product needs engineering
Construction

• Taking the prototypes (or learning from them) and creating a whole
• Quality must be attended to: usability (of course), reliability, robustness, maintainability, integrity, portability, efficiency, etc
• Product must be engineered
  Evolutionary prototyping
  ‘Throw-away’ prototyping
Overview

- Prototyping and construction
- **Conceptual design**
- Physical design
- Generating prototypes
- Support for design
Conceptual design: from requirements to 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”
• Don’t move to a solution too quickly. Iterate, iterate, iterate
• Consider alternatives: prototyping helps
Is there a suitable metaphor?

• Interface metaphors combine familiar knowledge with new knowledge in a way that will help the user understand the product.

• Three steps: understand functionality, identify potential problem areas, generate metaphors

• Evaluate metaphors:
  How much structure does it provide?
  How much is relevant to the problem?
  Is it easy to represent?
  Will the audience understand it?
  How extensible is it?
Considering interaction types

• Which interaction type?
  How the user invokes actions
  Instructing, conversing, manipulating or exploring

• Do different interface types provide insight?
  WIMP, shareable, augmented reality, etc
Overview

- Prototyping and construction
- Conceptual design
- Physical design
- Generating prototypes
- Support for design
Expanding the conceptual model

• What functions will the product perform?
  What will the product do and what will the human do (task allocation)?

• How are the functions related to each other?
  Sequential or parallel?
  Categorisations, e.g. all actions related to telephone memory storage

• What information needs to be available?
  What data is required to perform the task?
  How is this data to be transformed by the system?
Using scenarios in conceptual design

- Express proposed or imagined situations
- Used throughout design in various ways
  - scripts for user evaluation of prototypes
  - concrete examples of tasks
  - as a means of co-operation across professional boundaries
- Plus and minus scenarios to explore extreme cases
Overview

- Prototyping and construction
- Conceptual design
- Physical design
- **Generating prototypes**
- Support for design
Generate storyboard from scenario

1. Thomson family gather around
2. System suggests Flotilla
3. System shows descriptions
4. System asks for details
5. Summary printed
Generate card-based prototype from use case
Overview

• Prototyping and construction
• Conceptual design
• Physical design
• Generating prototypes

• Support for design
Support for design

- Patterns for interaction design
  - individual patterns
  - pattern languages
  - pattern libraries
- Open source systems and components

- Tools and environments
Summary

- Different kinds of prototyping are used for different purposes and at different stages
- Prototypes answer questions, so prototype appropriately
- Construction: the final product must be engineered appropriately
- Conceptual design (the first step of design)
  - Consider interaction types and interface types to prompt creativity
- Storyboards can be generated from scenarios
- Card-based prototypes can be generated from use cases