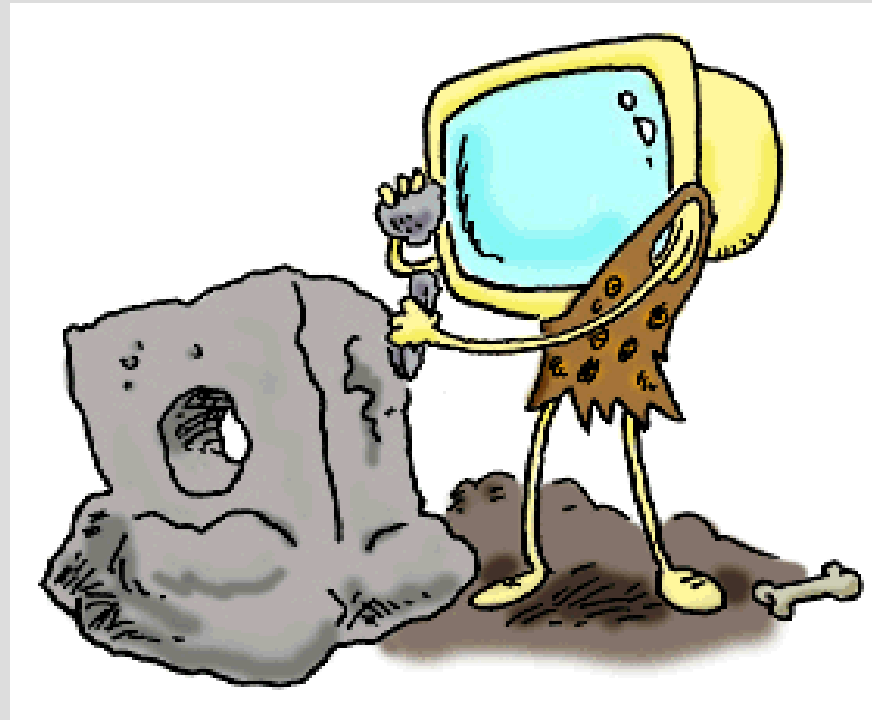


Design, Prototyping & Construction



Overview

- Prototyping & construction
- Conceptual design
- Physical design
- Tool support



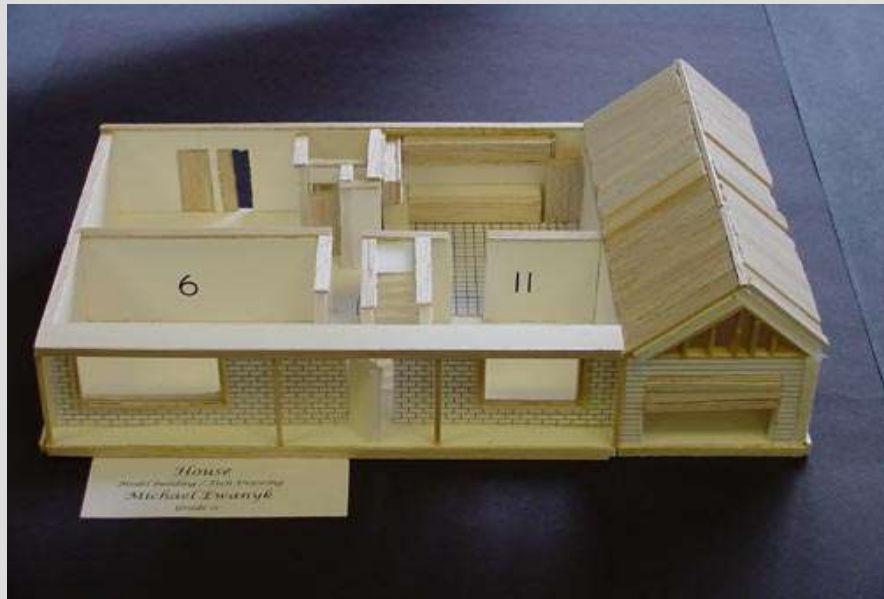
Prototyping & construction

- 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



What is a prototype in HCI ?

- It can be (among other things):
 - A series of screen sketches
 - A storyboard i.e a cartoon like sequence of scenes
 - Slideshow
 - Video simulating the use of the system
 - A lump of wood (Palm Pilot)
 - A cardboard mock-up
 - A piece of software with limited functionality written in the target language or another language

Why prototype ?

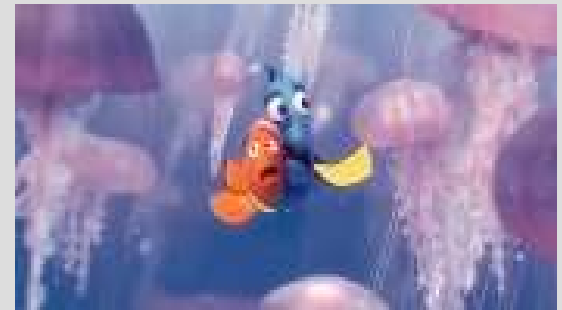
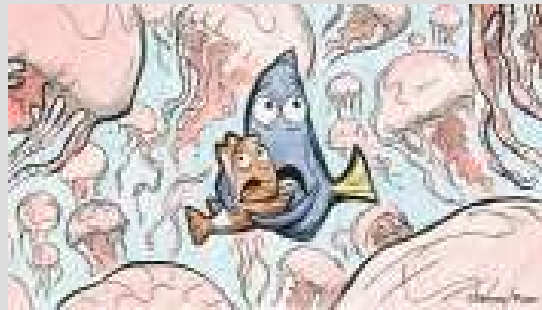
- Evaluation and feedback are central to ID
- Stakeholders can see, hold, interact with a prototype more easily than a document or a drawing
- Team members can communicate effectively
- You can test out ideas for yourself
- It encourages reflection – very important
- Prototypes answer questions and support designer in choosing between alternatives

What to prototype ?

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

Low Fidelity Prototyping

- Uses a medium that 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
 - Story boards
 - Wizard of Oz



The Palm Pilot story

- “Less is more”
- 1996: 0.5 million units in the first 7 months
- Apple Newton
 - Elegant, powerful
 - Failure
- Jeff Hawkins (founder)
 - Walked around with a block of wood in his pocket
- Concentrate on usability & essential functionality – every pixel counts
- No save button – 1 application at a time



Storyboards & sketching

- 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 the design
- Sketching
 - Important to low fidelity prototyping
 - Don't be inhibited about drawing ability.
Practice simple symbols

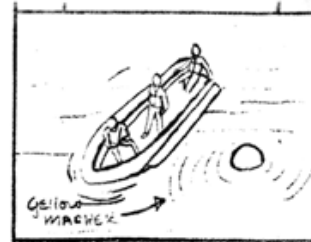
Storyboard



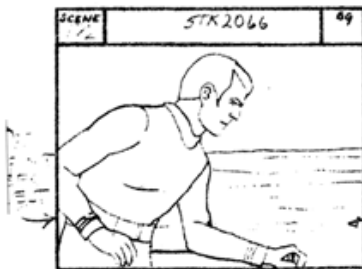
DOMAR
That is impossible. Only AQUAN
mutation is designated by the surge-
Records.



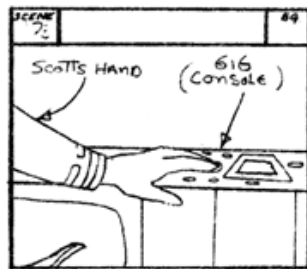
SPOCK & KIRK LOOK AT EACH OTHER



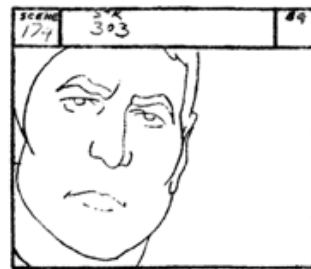
McCoy's VOICE
(on communicator)
Enterprise to Mr. Scott



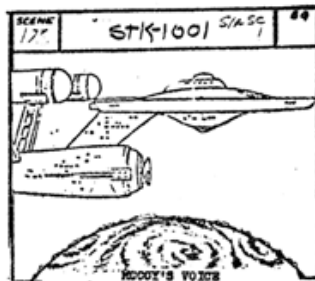
SCOTT
Scott here. What is it,
Doctor?



SCOTT'S HAND (616 console)
SCOTT'S VOICE
(on communicator, urgently)
Sensors just gave us an update
reading, Scotty. There's a sea
quake due in that area -- a bad
one. Complete topography changes



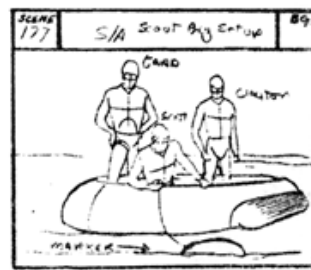
SCOTT
How soon?



SCOTT'S VOICE
(on communicator)
Within four hours. When are
Jim and Spock due to make contact?
SCOTT
MCCOY'S VOICE



SCOTT
We can try like blue blazes.
Scott out.



(EMERGENCY SIGNAL BUZZ)

DASS

Participatory case study

- Automatic library checkout system
 - Lookup book on computer
 - Note down the locator
 - Scan barcode of book
 - Green light indicates book has been scanned

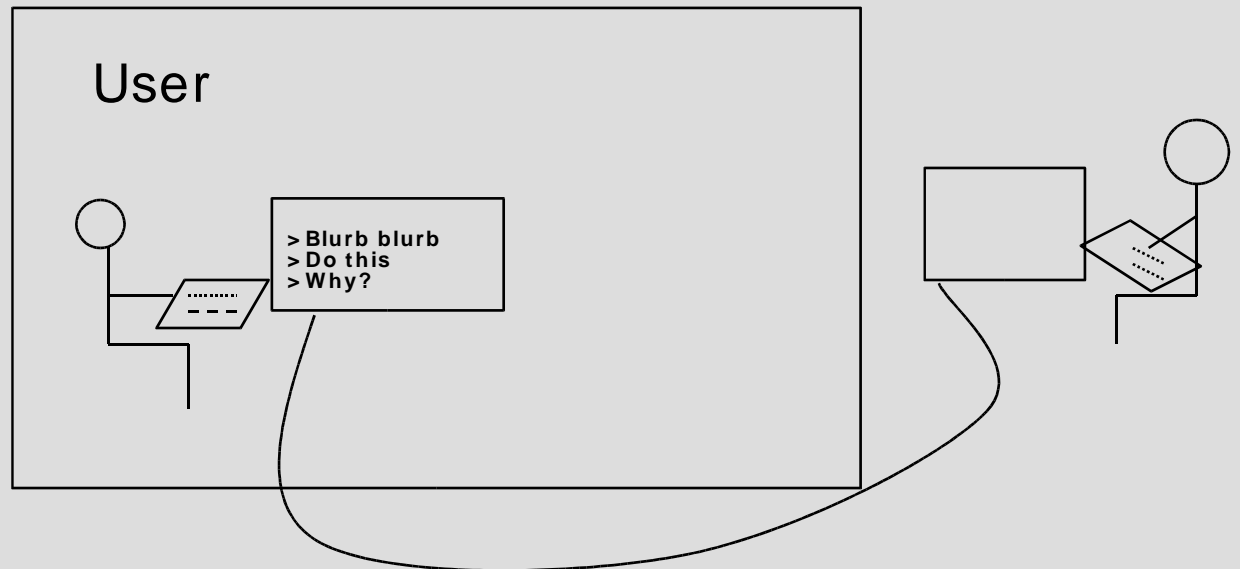
Using index cards

- Index cards (3x5 inches)
- Each card represents one 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 the system
- Usually done early in design to understand users expectations
- Applications ?
- Cons ?



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, Smalltalk
- Danger that users think they have a full system ... compromises

Vertical vs Horizontal Prototyping

- Where the compromises are made ?
- For software-based prototyping maybe there is a slow response ? sketchy icons ? limited functionality ?
- Horizontal
 - Wide range of function but with little detail
- Vertical
 - A lot of detail for only a few functions
- Compromises can't be ignored – importance of engineering

Construction

- Taking the prototypes (or learning from them) and creating a whole
- From design to implementation
 - Evolutionary prototyping
 - Throwaway prototyping
- Quality must be attended to: usability, reliability, robustness, maintainability, integrity portability, efficiency
- “Software is the only engineering field that throws together prototypes and then attempts to sell them as delivered goods” Constantine & Lockwood 1999

Conceptual Design: from requirements to design

- Transform user needs/requirements 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

Three perspectives for a conceptual model

- 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

Three perspectives for a conceptual model

- Which interaction paradigm ?
 - Desktop paradigm with WIMP interface (windows, icons, menus, pointer)
 - Ubiquitous computing
 - Pervasive computing
 - Wearable computing
 - Mobile devices
 -

Three perspectives for a conceptual model

- 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 is it relevant to the problem ?
 - Is it easy to represent ?
 - Will the audience understand it ?
 - How extensible is it ?

Expanding the conceptual model

- What functions will the product perform ?
- What will the product do and what will the human ? (task allocation)
- How are functions related to each other
 - Sequential or parallel ?
 - Categorizations, e.g all functions related to editing
- What information needs to be available ?
 - What data is required to perform the task ?
 - How is this data transformed/represented 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
 - Everything great
 - Everything goes wrong

Using prototypes in conceptual design

- Allow evaluation of emerging ideas
- Low fidelity prototypes used early on, high fidelity prototypes used later

Physical Design

- Considers more concrete, detailed issues of designing the interface
- Iteration between conceptual and physical design
- Guidelines for physical design
 - Nielsen's heuristics
 - Shneiderman's eight golden rules
 - Style Guides: commercial, corporate
 - Decide “look & feel” for you
 - Widgets prescribed e.g icons, toolbar

Physical Design

- Different kinds of widgets (dialog boxes, toolbars, icons, menus etc)
 - Menu design
 - Icon design
 - Screen display
 - Information display

Menu Design

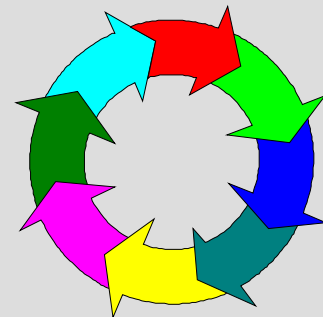
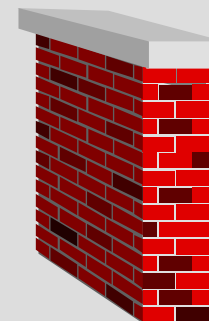
- How long is the menu to be ?
- In what order will the items appear ?
- How is the menu to be structured, e.g when to use sub-menus, dialog boxes ?
- What categories will be used to group menu items ?

Menu Design

- How will division into groups be denoted, e.g different colors, dividing lines ?
- How many menus will there be ?
- What terminology to use ? (results of the requirements activity will indicate this)
- How will any physical constraints be accommodated e.g mobile phone ?

Icon design

- Good icon design is difficult
- Meaning of icons is cultural and context sensitive
- Some tips:
 - Always draw on existing traditions or standards
 - Concrete objects or things are easier to represent than actions
 - From clip art, what do these mean to you ?



Screen Design

- Two aspects:
 - How to split across screens
 - Moving around within and between screens
 - How much interaction per screen ?
 - Serial or workbench style ?
 - Individual screen design
 - White space: balance enough information/interaction and clarity
 - Grouping items together: separation with boxes ?
Lines ? Colors ?

Screen Design: Splitting functions across screens

- Task analysis as starting point
- Each screen contains a single, simple step ?
- Frustration if too many simple screens
- Keep information available: multiple screen open at once

Screen Design: Individual Screen Design

- Draw user attention to salient point, e.g. color, motion, boxing
- Animation is very powerful but can be distracting
- Good organization helps: grouping, physical proximity
- Tradeoff between sparse population and over-crowding

Information Display

- Relevant information available at all times
- Different types of information imply different types of display
- Consistency between paper display and screen data entry (calendar, watch examples)

Tool Support

Myers, 1995

- Help design interface given specification
- Help implement interface given specification
- Create easy to use interfaces
- Allow designer to rapidly investigate different designs
- Allow non-programmers to design and implement user interfaces
- Automatically evaluate the interface and propose improvements
- Allow the end user to customize the interface
- Provide portability
- Be easy to use

Successes of UI tools

Myers, 2000

- Window managers and toolkits
- Event languages
- Interface builders (Visual Basic)
- Component systems (Direct X, JavaBeans)
- Scripting languages (Python, Perl)
- Hypertext and markup languages XML
- Object-oriented programming

Failures of UI tools

Myers, 2000

- User interface management tools:
 - Abstract away from manipulation of interface elements
- Formal language based tools
 - State-transition diagrams, context-free grammars
 - Better for dialog-based systems not direct manipulation
- Constraints
 - Too difficult to set-up, maintain
- Model-based and automatic techniques
 - Generate interface – only work in specific domains

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)
- Physical design (e.g. Menus, icons, screen design, information display)
- Prototypes and scenarios are used throughout design