User Interface Design

Ruchita Shah

UI Design

- Focus on three areas
  - I/f b/w s/w components
  - I/f b/w s/w & external entities
  - I/f b/w human & the computer
- Some i/f difficult to learn, use, confusing
- study the people who use the i/f

The Golden Rules

- Place the user in control
- Reduce User’s Memory Load
- Make the I/f Consistent

Place the user in control

- Most i/f constraints & restrictions for simple interaction
- such i/f easy to build but complex to use, to maintain control

1. Define interaction modes in a way that does not force user into unnecessary or undesired action - able to enter & exit current state with little or no effort
2. Provide for flexible interaction – different users have diff preferences, choice provided, interact via keyboard command, mouse, digitizer pen, voice recognition etc, use common sense
3. Allow user interaction to be interruptible & undoable – in sequence of actions, able to interrupt sequence & undo action

Place the user in control

4. Streamline interaction as skill levels advance & allow the interaction to be customized – repeated work, macro mechanism for advanced user
5. Hide technical internal from casual user – user not be aware of OS, file management functions or technicality, i/f never allow user inside the machine
6. Design for direct interaction with objects that appear on the screen – user feels control when manipulate objects to perform tasks, ex, stretch the size

Reduce User’s Memory Load

- more to remember, more errors, system should remember info & assist user

1. Reduce demand on short term memory – during complex task, short term memory in demand, reduce requirement to remember past actions & result, provide visual cues
2. Establish meaningful defaults – sensible initial defaults, reset option

Reduce User’s Memory Load

3. Define shortcuts that are intuitive – mnemonic for system functions meaningful
4. The visual layout of the i/f should be bases on a real world metaphor – same as real world layout
5. Disclose info in a progressive fashion – hierarchical organization, first at high level of abstraction, detail after user indicate interest

Make the I/f Consistent

- info in consistent manner
  - visual info according to design std, maintained in all screens
  - i/p limited & consistent throughout the application
  - consistent navigation

1. Allow user to put the current task into a meaningful context – provide indicators to know context of work, user able to determine from where he has com & alternatives

Make the I/f Consistent

2. Maintain consistency across a family of applications – same design rules
3. If past interactive models have created user expectations, do not make changes unless necessary – user adaptive to interaction sequence, change cause confusion

USER I/F DESIGN

- Begins with modeling of system functions
- Interactions required decided
- Design issue applied & tools used to create prototype & the actual

I/f Design Model

- 4 models to design i/f
  - s/w engineer creates design model
  - human engineer – user model
  - end user – system perception
  - implementer – system image
- each different
- i/f designer reconcile differences & derive consistent i/f

I/f Design Model

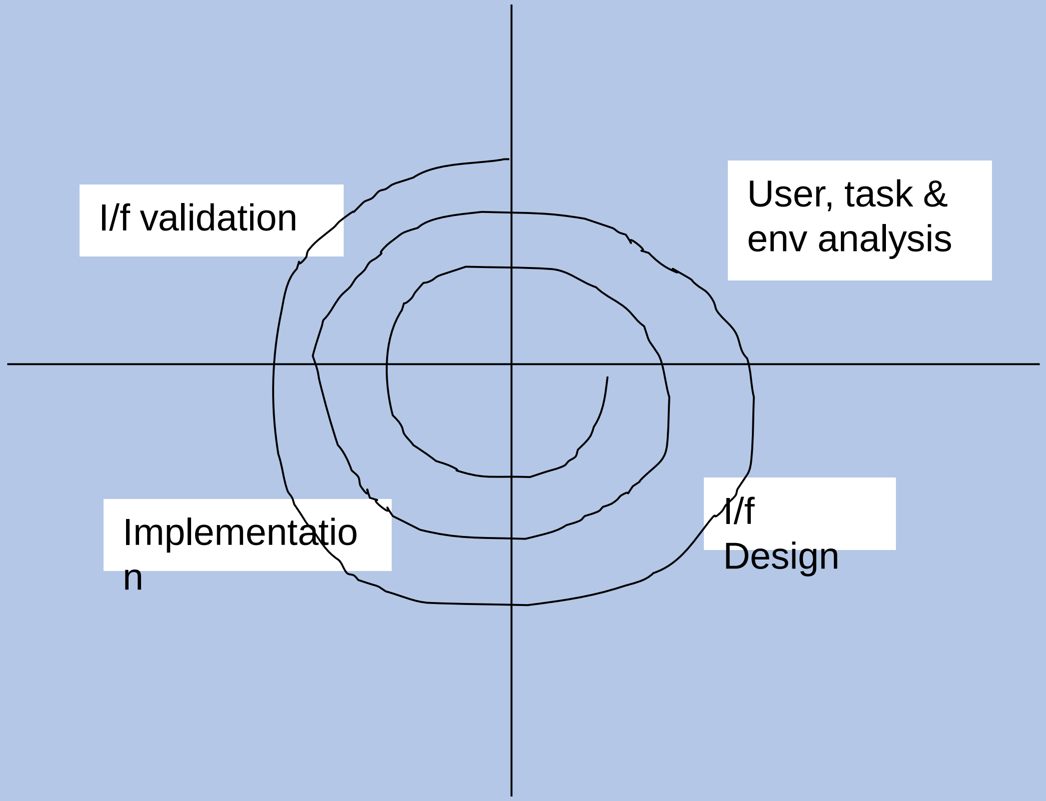
- To build effective i/f begin with understanding users, their age, abilities, education, background, goals & personality
- User fall to one of the categories
  - Novice – no syntactic knowledge, little semantic knowledge
  - Knowledgeable, intermittent users – reasonable semantic knowledge, low syntactic info
  - Knowledgeable, frequent user – good semantic & syntactic knowledge, looks for shortcuts & abbreviations

I/f Design Model

- System perception – image of system by end-users
- system image combines outward manifestation & supporting info describing system syntax
- If system image & perception coincide, user comfortable with such s/w
- Design models to accommodate user model, system image & system perception

The User I/f Design Process

- iterative process, spiral model, 4 framework activities
  - User, task & env analysis & modeling
  - I/f design
  - I/f construction
  - i/f validation



The User I/f Design Process

- Analysis focus on profile of users, their skill level, business understanding, etc recorded, categorize, requirements elicited, then task analysis, tasks to accomplish goal of system identified, described & elaborated
- User env analysis focus on physical work env
  - Where i/f will be located
  - Will the user sitting, standing or performing other tasks unrelated to i/f
  - Does the i/f h/w accommodate space, light or noise constraints
  - Are there special human factors considerations driven by env factors

The User I/f Design Process

- I/f design define set of i/f objects & actions to perform tasks & usability goal
- Implementation begin with creation of prototype
- Validation focus on
  - Implementation of every user task correctly, achieve all user requirements
  - Degree of easiness of using & learning i/f
  - User’s acceptance

Task Analysis & Modeling

- To understand the tasks currently performed by the user : Map into similar set of tasks in context of user i/f
- Another method : by observing & studying existing system
- Derive set of user tasks to accommodate user model, design model & system perception
- First step is to define & classify the tasks

Task Analysis & Modeling

- one of two approaches
  - Step wise elaboration : observes work, find major activities, elaborated in sub activities, model consistent with user model & system perception
  - OO approach : observes physical objects used by users & action applied to each object

Interface Design Activities

- After defining each tasks, design begins
- Steps for i/f designs are :

1. Establish goals & intention for the tasks
2. Map each goal & intention to a sequence of specific action
3. Specify the action sequence as it will be executed at i/f level

Interface Design Activities

4. Indicate the state of system during action
5. Define control mechanism to alter the system state like cancel, abort
6. Show control mechanism effect
7. Indicate user interpretation

Defining I/f Objects & Actions

- User scenario parsed
- Description of user scenario
- Objects & actions isolated & listed
- Categorized by type, target, source & application objects
    - object – asource object – dragged & dropped onto a target object ex report icon on printer icon result in print out
    - application pplication specific data, not directly manipulated ex mailing list, Next screen layout is performed
- An interactive process
- Graphical design & placing of icons
- Definition of screen text, specification, titling of windows, definition of menu items etc,

Design Issues

- 4 issues
  - System response time
  - User help facility
  - Error handling
  - Command labeling
- considered in beginning
- easy change & low cost

Design Issues

- System response time
  - Primary complaint for interactive systems
  - 2 characteristics, length & variability
  - Too long response time creates frustration
  - very brief response time may force user to rush & make mistakes
  - Avg response time must not vary
  - Establish interaction rhythm

Design Issues

- User help
- Must for interactive systems
- on-line help to resolve difficulties without leaving i/f
- 2 types of help
  - Integrated - integrated context specific, reduces time of search & increases friendliness
  - Add-on online help with limited query capabilities, long search

Design Issues

- Design issues to be addressed are :
  - Will help available for all functions & at all times
  - How will user request help – menu, function key, command
  - Help representation – separate window, reference to printed document, one – two line suggestion in a fixed location
  - How to return back to normal interaction – return button, function key
  - Information structure – flat structure, layered hierarchy

Design Issues

- Error message & warnings:
- Must produce proper information
- Misleading info increases user frustration
- Should have following characteristics :
  - Massage describe the problem in user understandable manner
  - Provide constructive advice for recovering
  - Indicate negative consequences of error
  - Audible or visual cue, beep, color, flash
  - Nonjudgmental error message
- Effective error message improve quality of interactive system, reduce user frustration

Design Issues

- Command Labeling:
- Before few years typed command
- Now window-oriented
- Many options like command oriented mode
- Issues are
  - command for menu options
  - form of command, control key, function key, typed command etc
  - difficult to learn & remember, if forgotten
  - customized or abbreviated command
- Common commands for different applications

Implementation Tools

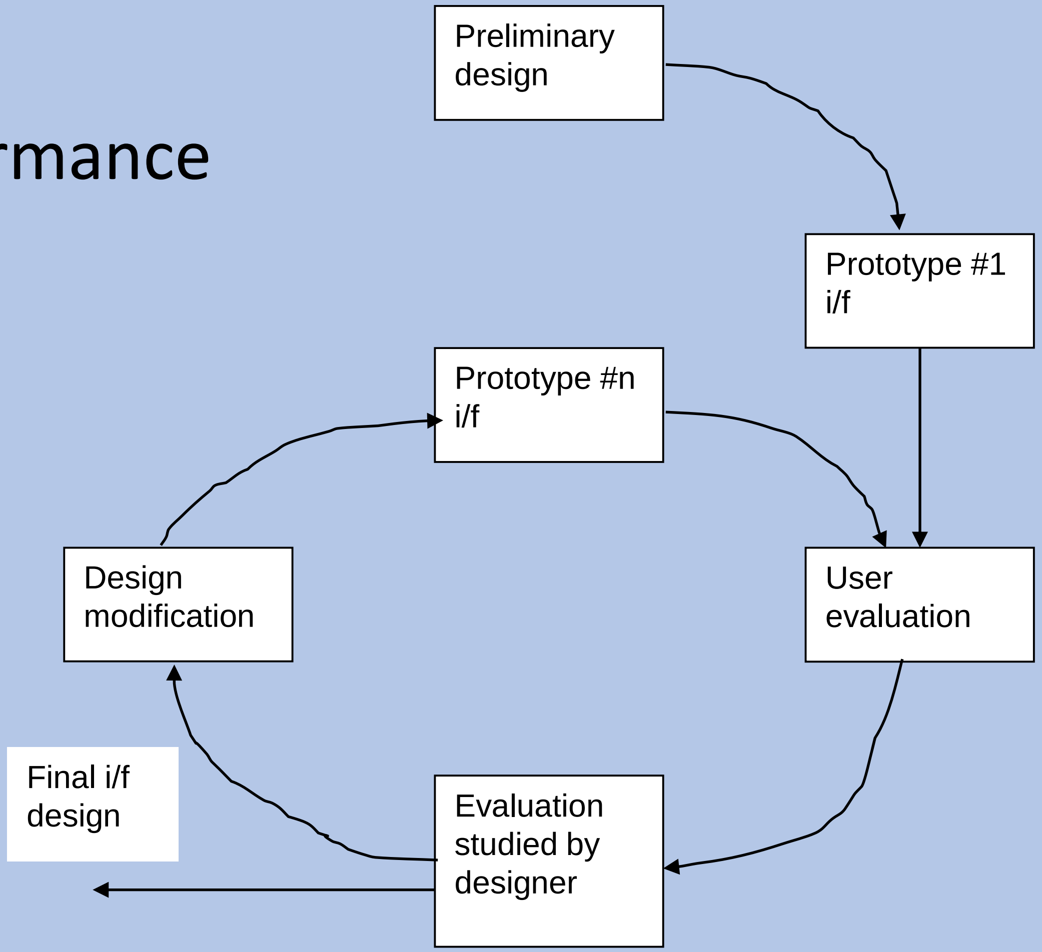
- Design model implemented as a prototype
- Examined & modified in iterative manner
- Class of i/f design & implementation tools
- User-interface toolkits or user-interface development system
- provide objects to create windows, menu, device interaction, error message, commands etc,

Implementation Tools

- These tool can
    - manage i/p devices
    - validate user i/p
    - handles error & display error messages
    - provide feedback
    - provide help
    - handle windows, scrolling
    - establish connection b/w application s/w & i/f
    - allow user to customize i/f

Design Evaluation

- After creating prototype
- Evaluate for requirements conformance
- Prototype approach is effective
- Have long development cycle



Design Evaluation

- Evaluation criteria applied in early phases such as :
- Length & complexity of written specification – learning required
- Number of user tasks & avg no of actions – interaction time & efficiency of system
- Number of actions, tasks & system states – memory load of users
- I/f style, help, error handling – complexity of i/f

Design Evaluation

- Qualitative & quantitative data collected & used to evaluate i/f
- Questions are distributed to users – yes/no type, numeric response, scaled, percentage etc
- ex. Were icons self explanatory, how easy to learn basic system operations etc
- Quantitative data collected by observing users during interaction, data
  - Number of tasks completed over time period
  - Frequency if action
  - Number of errors
  - Type of errors etc