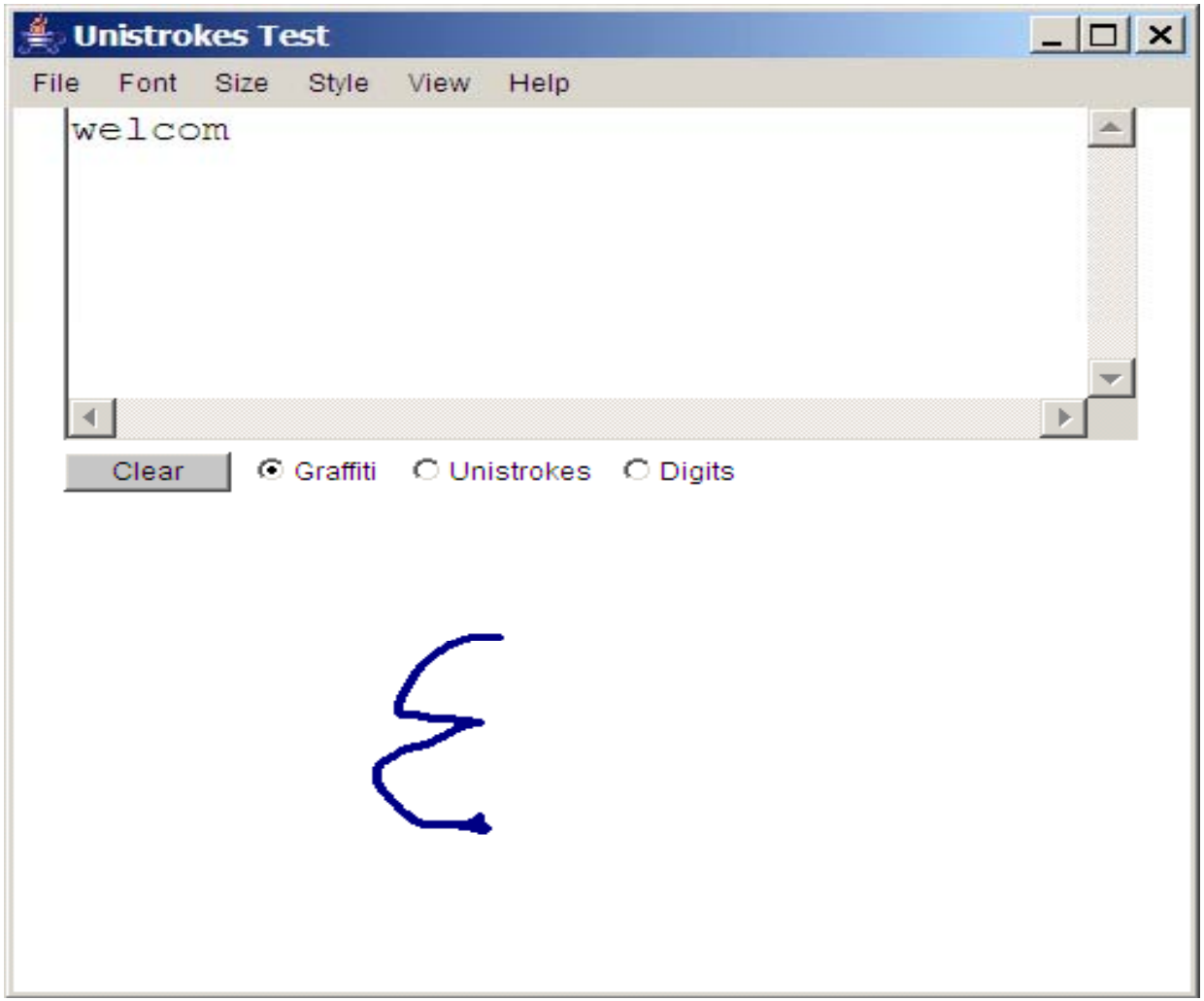
CSE 3461 User Interfaces

Introduction

Instructor (Section A):
Scott MacKenzie

Welcome Demo



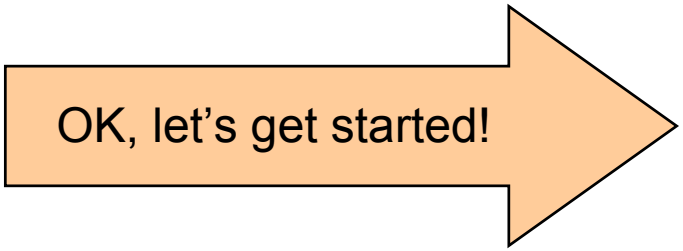
Course Web Page

- URL

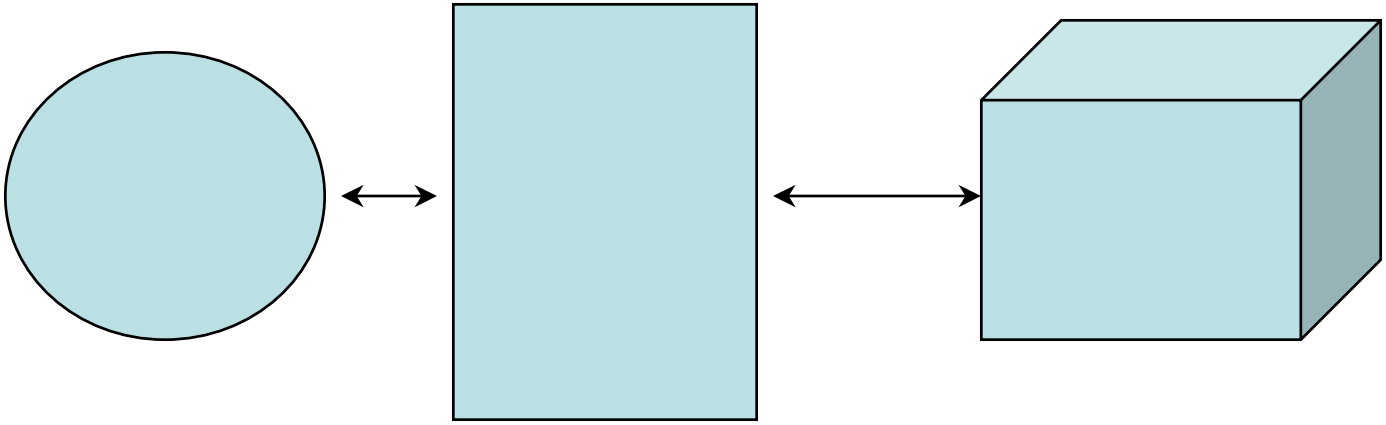
http://www.cse.yorku.ca/course/3461/

Course Contents and Requirements

- (view course web site with browser)



Simple View of the User Interface



A user interface is the junction between the user and the computer

Correct View

- A User Interface is much more than a layer slapped between the user and the computer
- Incorporates an understanding of
  - the user
  - the task and task flow
  - the environment
  - etc.

Example Programs

DemoLargestConsole.java

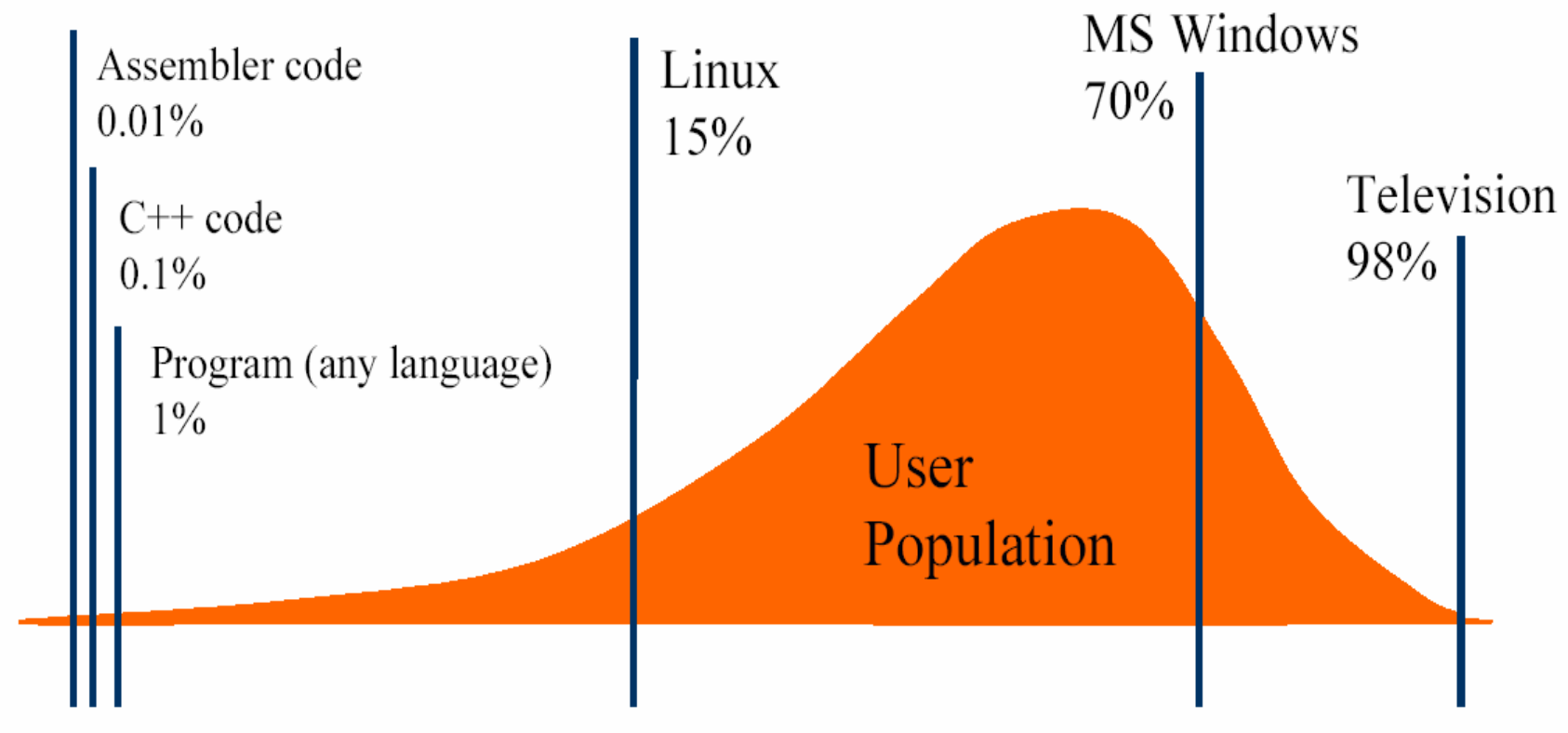
DemoLargestGUI.java

Demo programs will be posted on the course web page shortly after each lecture.

User

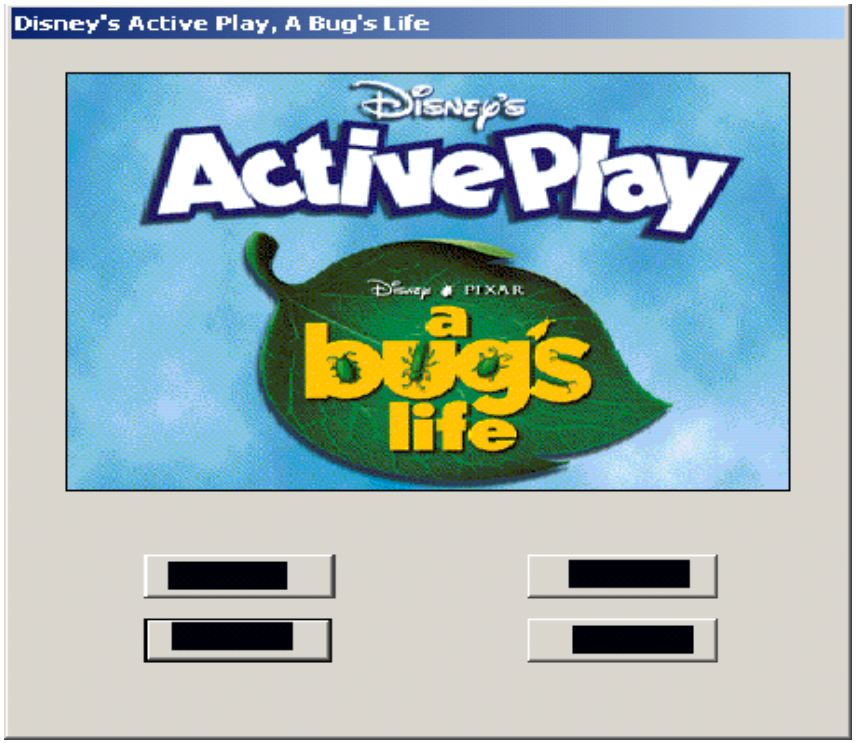
- Issues involved in designing a user interface for even a simple task can be complex
- There are lots of ways to design an interface to do the same task (we just saw some examples)

Usable for Whom?



Each example is “usable” by that portion of the user population to the left of black line.

Possible Usability Issues?

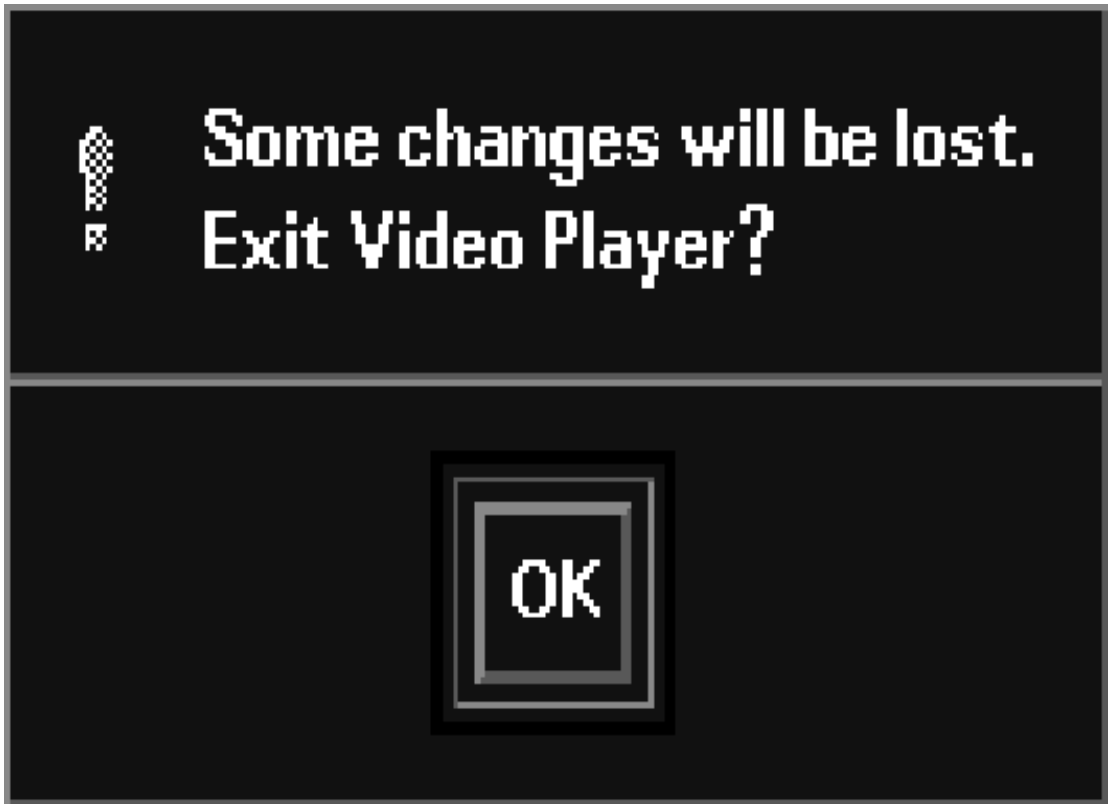


How it appears to a child

Bug's Life Example

- Consider who the users are
- Breaks user expectation (must launch the program to uninstall it)
- User: "I find myself having to reinstall this program every other week!"

Good or Bad?



How do you cancel?

What is the User Interface (UI)?

- Different actors perceive it very differently!
- **Developer's View**
  - Application functionality is separate from UI
  - UI often seen as an add-on
- **User's View**
  - Users don't distinguish between the UI and the entire application
  - If the UI is well designed and usable, then entire application appears usable

Credo: The interface IS the computer!

What is the UI? (continued)

- UI includes all aspects of the system that influence interaction with users
- This includes:
  - conceptual model (how UI objects correspond to real-world objects)
  - a metaphor, to help the user (e.g., “desktop”)
  - input controls and their behaviour
  - means for navigation within and flow between screens
  - Integration and consistency among different applications
  - visual design of the screens

Technology Trends

- Trends in technology have increased the need for improved UIs and UI design practices:
  - Network/distributed systems allow access in remote locations, or across an enterprise
  - Greater memory and faster processing are available at ever decreasing costs
  - More people have access to computing power
  - People are “on the move” (mobile computing)
- Technology now exists for improved UIs and UI designs

New Technology is NOT Enough!

- New interface technology alone does not produce usable interfaces!
  - Graphical user interfaces (GUIs) are not intrinsically more usable than traditional console applications
  - GUIs can be less usable if they are poorly designed
- Usable interfaces require good design and a great deal of effort in their manufacture

User Interfaces: Code/Cost/Effort

- Different Statistics
  - UI is 47% - 60% of the total code
  - The UI is minimally 29% of the software development project budget
  - The UI may take as much 40% of the development effort

Interesting Stat

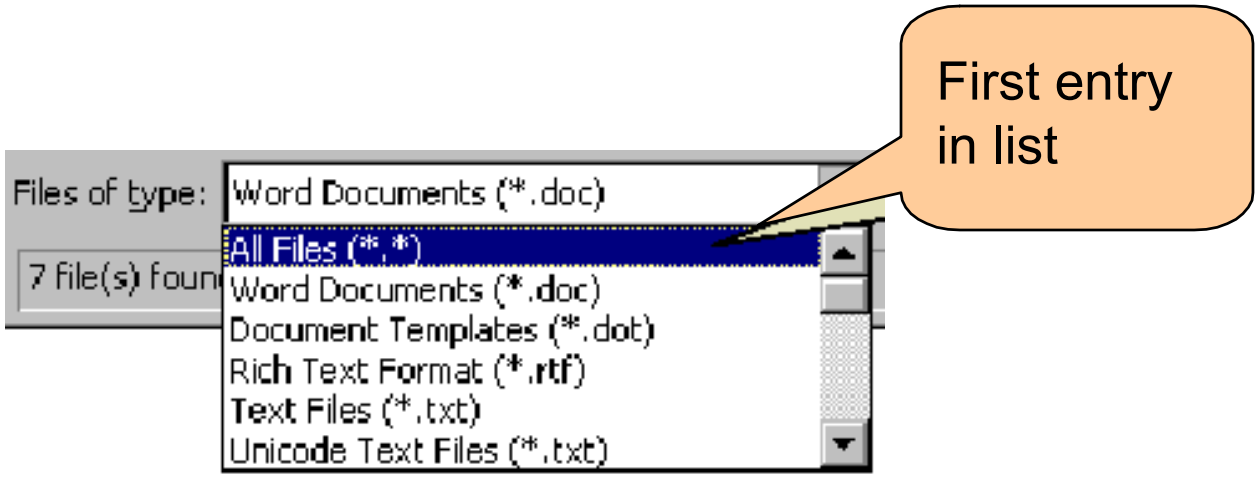
- 80% percent of software life cycle costs occur after the product is released, in the maintenance phase
  - Of that work, 80% is due to unmet or unseen user requirements
  - Only 20% of this is due to bugs or reliability problems
- (Source: Karat, C. Usability engineering in dollars and cents. *IEEE Software*, May 1993, p 89.)

High Quality UIs are in Demand

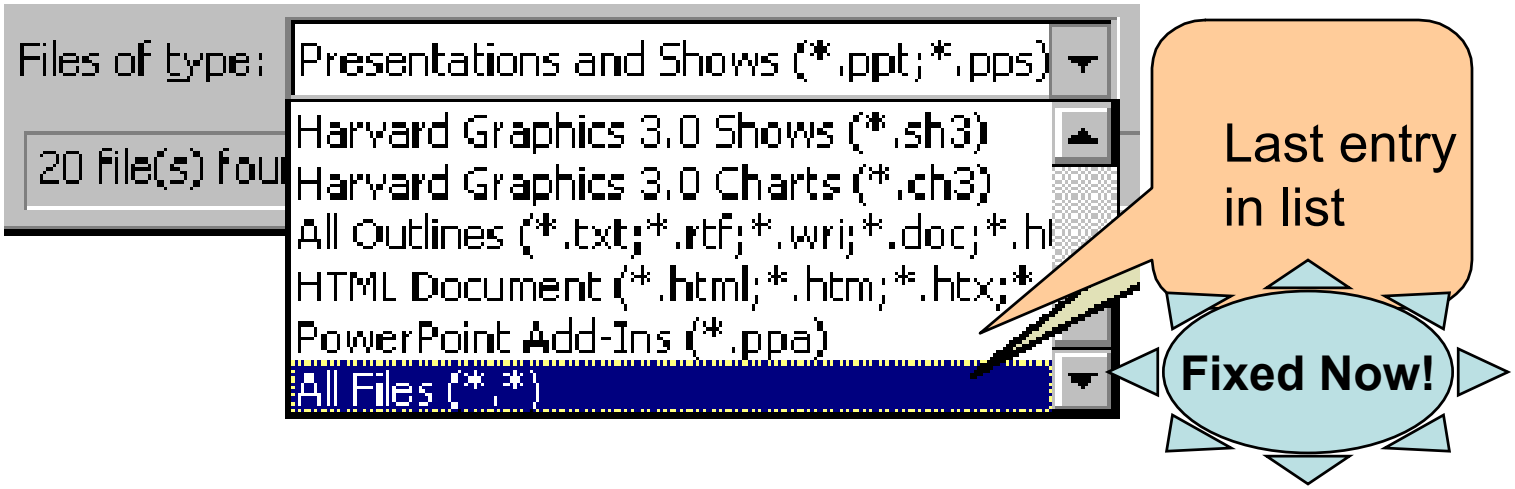
- A good UI is vital for a quality software product
  - UIs that do not work well make it impossible for users to access the functionality they require; i.e., users become less productive
- Human capital is more expensive
- Therefore need UIs and systems that are:
  - Learnable, efficient, memorable
  - Not error prone
  - Satisfactory for the user

Example for Bad UI

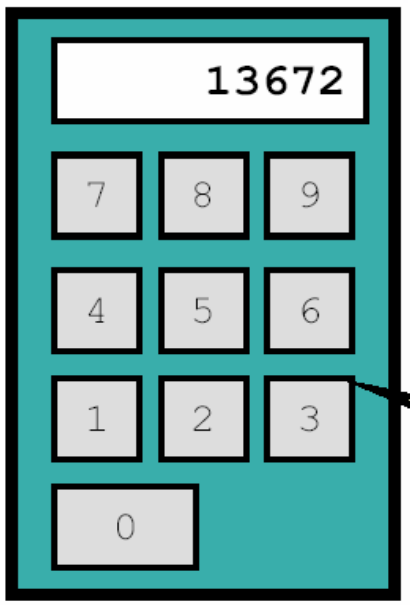
Word:

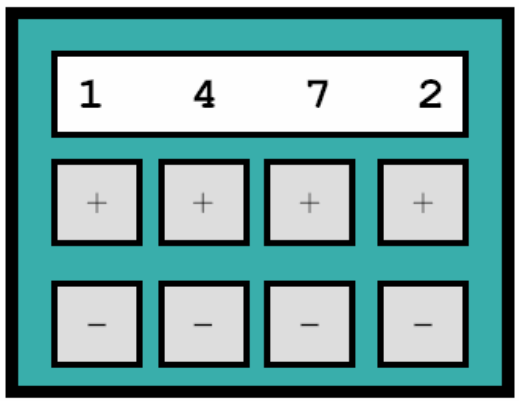


Power Point:



Example for Bad UI (2)





An accidental key press and the dose can be out by a factor of ten!

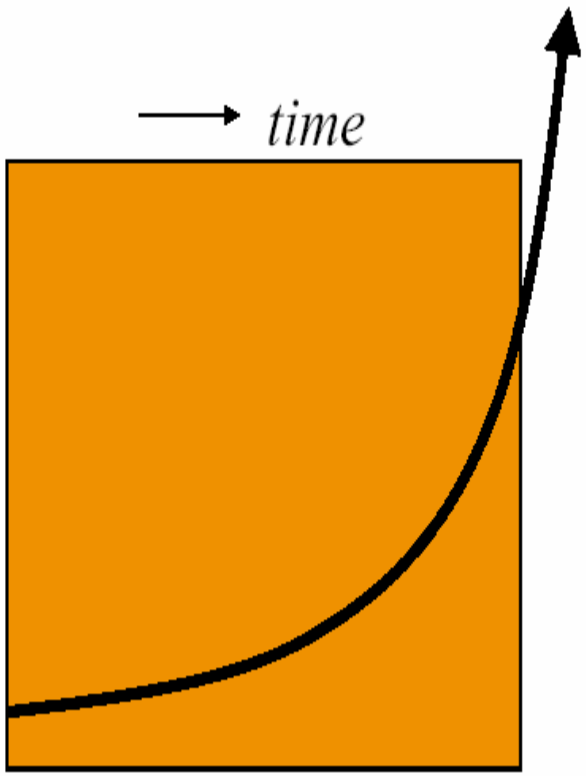
Bad UIs Lead to Major Problems

- A $3 million application for an insurance company to be used by independent agents to support them in selling their company's products. However, agents refused to use the application because the system was "un-learnable" and "unusable".
- In a customer service organization, training on the system took 6 months, but employees typically stayed only 18 months in that department.
- Extensive and expensive functionality in a Human Resources system was not used because users forgot how to access it one week after training.

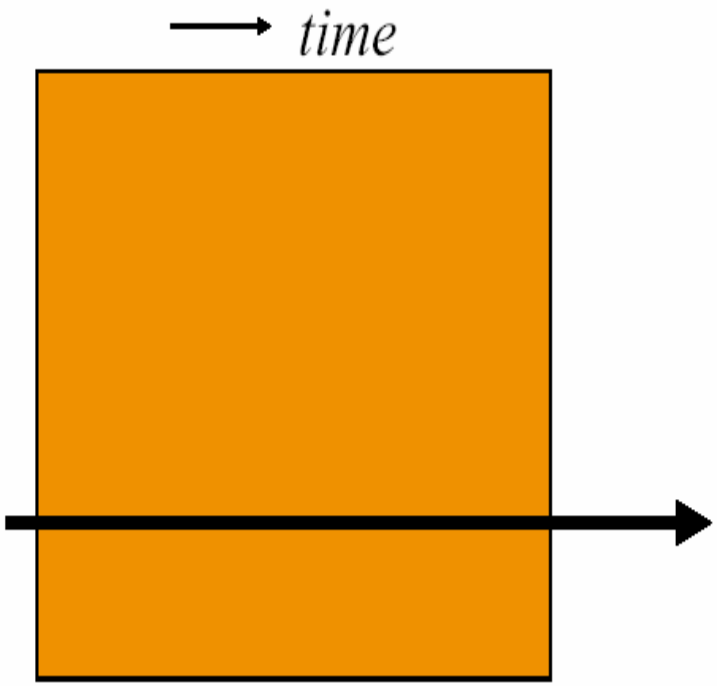
Good Designs…

- Work the way the user thinks they should (intuitive)
  - Allows the user to focus on task at hand and not worry about the underlying technology and interaction methods
- Easier to learn and use than previous technology
- More efficient than manual systems
- Minimize user errors
- Promote user satisfaction (users should feel that they are accomplishing *more* with the system than without the system)
- Lower support costs
- Transfer skills across applications
- Fully utilize system functionality

Don Norman's Observation



Computer Processing Power

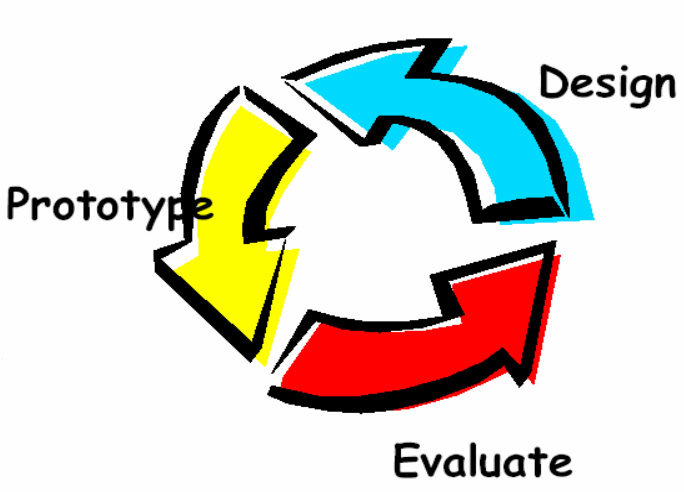


Human Processing Power

Human-Computer Interaction

- Human-computer interaction is a discipline concerned with the design, implementation, and evaluation of interactive systems for human use and with the study of major phenomena surrounding them.

The HCI lifecycle is an iterative cycle that involves designing and evaluating with “users” as much as possible.



HCI is Multidisciplinary

- Human side:
  - Cognitive psychology
  - Ergonomics and human factors
  - Sociology and anthropology
  - Linguistics
  - Communication theory
  - Social and organizational psychology
  - Graphic and industrial design

HCI is Multidisciplinary (2)

- Machine side:
  - Computer science
  - Engineering
  - Computer graphics
  - Operating systems
  - Programming languages
  - Software engineering
  - Development environments
  - Artificial intelligence

Good are UIs Hard to Build

- Multiprocessing requirements
  - Multiple inputs, redraws
  - Synchronization, threads, deadlock prevention
- Need abort, undo, redo
  - Requires lots of state information to be kept
- Real-time requirements
- Must be robust
  - Users do lots of odd things!

Good are UIs Hard to Build (2)

- API & UI logic complexity
- Reactive instead of proactive
  - User dictates what the system should do
- Hard to modularize (OO design helps)
- Exhaustive testing of UIs is hard – how to ensure robustness?
- Evaluation with users is time consuming

Where do we go from here?

- Implementation
  - Java JFC/Swing programming
  - Widgets, UIMS
  - Event-driven programming
  - Model-View Controller
- Human-Computer Interaction (HCI)
  - User interface design

Implementation

- Console applications (CUI's)
  - Command-line and natural language interfaces
- Interaction devices (input & output devices)
  - User interfaces for virtual environments
- Graphical User Interfaces (GUI's)
  - Window systems, toolkits, frameworks
  - GUI builders
- Mechanics
  - Event-driven programming
  - Model-View-Controller architecture

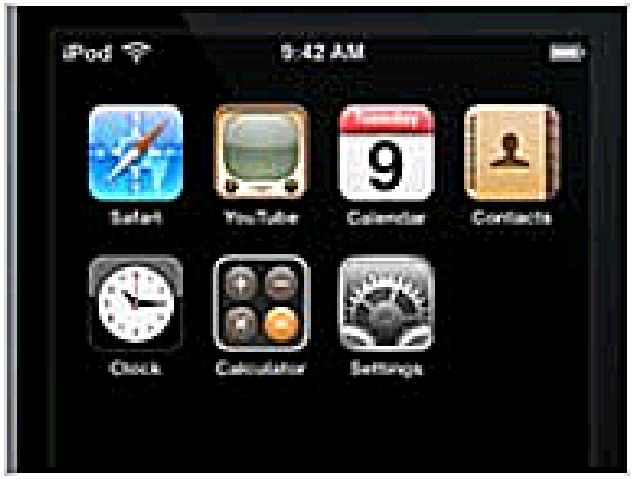
Advanced Design (COSC 4441)

- Designing with users
  - Focus on how users work
  - User-centred design
  - Participatory design
- Designing without users
  - Task-centred design
  - UI design notations
  - Usability heuristics
  - Guidelines

Evaluation (COSC 4441)

- Evaluation with user testing
- Evaluation with usability inspection methods
  - heuristic evaluation
  - cognitive walk-through

In Yesterday's News



Apple adds touch screen to iPods

Apple has refreshed its iPod lineup, adding touch screen and wireless internet connectivity to the devices. **more »**

- UI issues for touch screen
  - How many fingers?
  - Pressure thresholds (hysteresis)
  - Taps vs. gestures
  - Gesture recognition
  - Performance benefits or deficits
  - User learning and retention

Thank You