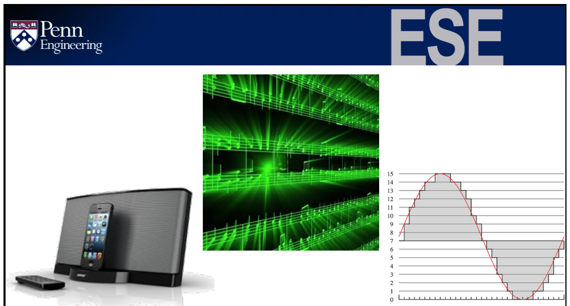


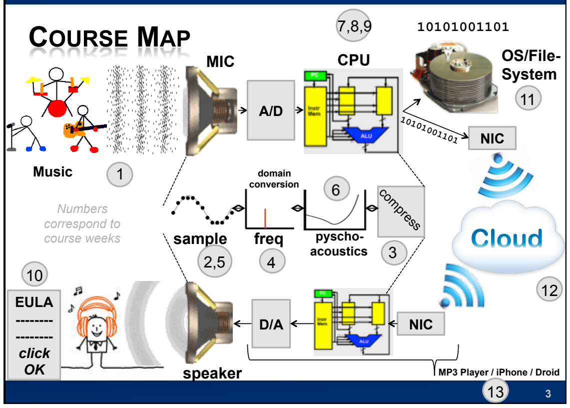
Lecture #13 – User Interface

ESE 150 – Digital Audio Basics

Lecture Topics

- Where are we on course map?
- User Interface
  - Motivation
  - Issues and Principals
  - Developer vs. User
  - Design Choices
  - Approaches and Prototyping
  - Enabling Technology
- Final

Course Map



User Interface

- When a user sees a product
  - See the interface
  - Not the underlying design
    - ….and that's the way it should be
- Interface determines if the user can get job done
  - …or will walk away frustrated
- Successful interface
  - Make it easy, pleasant to use
  - Hide all the complexity that makes it work

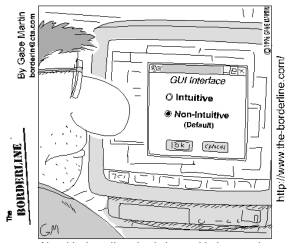
Dilbert Diagnosis

- http://dilbert.com/strip/2002-09-23
- http://dilbert.com/strip/2002-09-24

Self Awareness

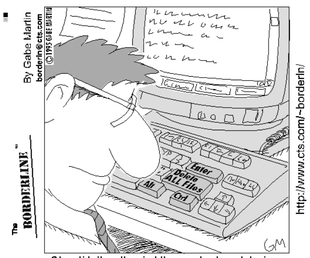
- I'm an Engineer
- I have a different perspective and understanding of technology than lay public
- My view of what's obvious/non-obvious probably not representative of intended user base
- …how do I (or team I'm in) compensate for that?
- This lecture, I'm talking about my weakness
  - And need for help
  - Not my strength
  - Won't do justice with solution…but maybe in raising issues, need for help
- Nonetheless, frustrated by bad design from others as much as anyone else…
  - Want "us" to do better.

The Problem is that Humans are Half of the User Interface



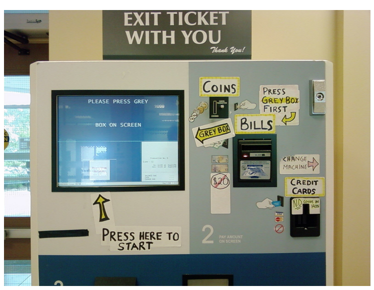
Al suddenly realizes that he's stumbled across the Mother of All undocumented Windows options.

And Even if Intuitive and Clear, User Interface Might not be so Good.



Stan didn't really mind the new keyboard design, but he just wished they hadn't put that new key so close to the "Enter".

And it's not just about our Workstation Interfaces…



Who's to Blame for Usability Failures?

- **Most Returned Products Work Fine:** Study Says Only 5 percent of returned products are genuinely defective: Yardena Arar, *PC World*, June 2, 2008 4:00 pm
- **Only 5 percent of consumer electronics products returned to retailers are malfunctioning** --yet many people who return working products think they are broken, a new study indicates.
- The report by technology consulting and outsourcing firm Accenture pegs the costs of consumer electronics returns in 2007 at **$13.8 billion** in the United States alone, *with return rates ranging from 11 percent to 20 percent*, depending on the type of product.

UI Examples: BAD

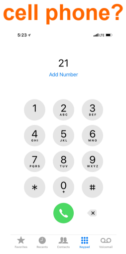
- Examples of infuriating / bad UIs?

UI Examples: Good

- Examples of pleasant/good UIs?

Phone

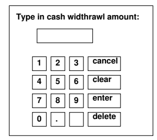
- **How is a cell-phone dialing interface better than a conventional POTS phone?**
- **…and how often do you dial on a cell phone?**
  - Alternative? Better?
- **How cell phone interface worse?**

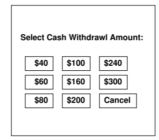


13

Preclass 2

- **Which interface easier? Why?**
  - Limit to vend $20, $300/day





14

Hot/Cold Water Interface

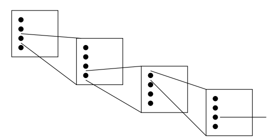
- Old: two knobs
  - Hot
  - Cold
- Newer: one knob
  - Tune heat
  - (maybe also volume)

Why built this way?

15

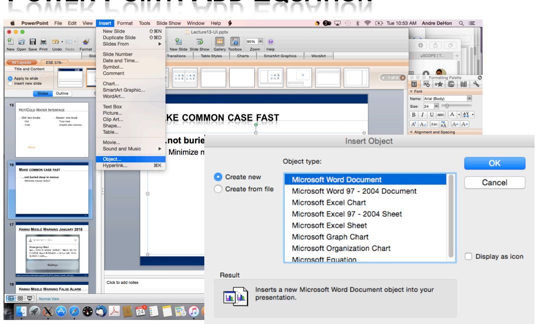
Make common case fast

- **…not buried deep in menus**
  - Minimize mouse clicks?



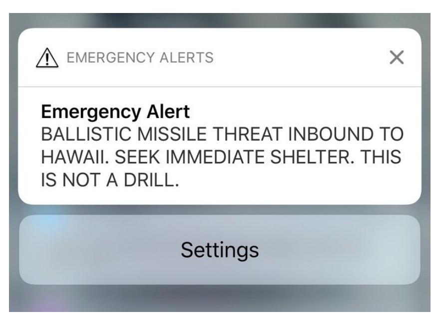
16

Power Point: Add Equation



17

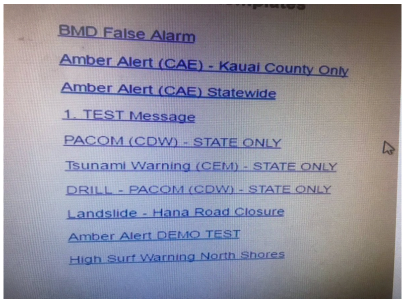
Hawaii Missle Warning January 2018



https://commons.wikimedia.org/wiki/File:2018_Hawaii_missile_alert.jpg

18

Hawaii Missle Warning False Alarm



https://www.theverge.com/2018/1/16/16896368/hawaii-false-missile-alert-system-confusing-interface-poor-design 19

Issues to be concerned with?

20

Donald Norman: UI Guru

Referring to Norman's book: *Design of Everyday Things*

- **Visibility –** visible functions aid user awareness; invisible functions are more difficult to find and know how to use.
- **Feedback –** return information about what action has been done and what has been accomplished.
- **Constraints –** restricting the kind of user interaction that can take place at a given moment.
- **Mapping –** the (functional, geometric, appearance) relationship between controls and their effects in the world.
- **Consistency –** use similar operations and use similar elements for achieving similar tasks.
- **Affordance –** an attribute of an object that allows people to know how to use it.

http://twobenches.wordpress.com/2008/06/05/don-normans-design-principles/ 21

Constantine and Lockwood

- **The structure principle.** Your design should organize the user interface purposefully, in meaningful and useful ways based on clear, consistent models that are apparent and recognizable to users, putting related things together and separating unrelated things, differentiating dissimilar things and making similar things resemble one another. The structure principle is concerned with your overall user interface architecture.
- **The simplicity principle.** Your design should make simple, common tasks simple to do, communicating clearly and simply in the user's own language, and providing good shortcuts that are meaningfully related to longer procedures.
- **The visibility principle.** Your design should keep all needed options and materials for a given task visible without distracting the user with extraneous or redundant information. Good designs don't overwhelm users with too many alternatives or confuse them with unneeded information.
- **The feedback principle.** Your design should keep users informed of actions or interpretations, changes of state or condition, and errors or exceptions that are relevant and of interest to the user through clear, concise, and unambiguous language familiar to users.
- **The tolerance principle.** Your design should be flexible and tolerant, reducing the cost of mistakes and misuse by allowing undoing and redoing, while also preventing errors wherever possible by tolerating varied inputs and sequences and by interpreting all reasonable actions reasonable.
- **The reuse principle.** Your design should reuse internal and external components and behaviors, maintaining consistency with purpose rather than merely arbitrary consistency, thus reducing the need for users to rethink and remember.

http://www.ambysoft.com/essays/userInterfaceDesign.html 22

Interaction Styles

- **Direct manipulation**
- **Menu selection**
- **Form fill-in**
- **Command language**
- **Natural language**

Ian Sommerville: Software Engineering, 7th Ed., 2004 23

Interaction Styles

Style	Main Advantages	Main Disadvantages	Applications
Direct manipulation	Fast and intuitive interaction; easy to learn	Only suitable where there is a visual metaphor for tasks and objects	Video games; CAD systems
Menu selection	Avoids user error; little typing required	Slow for experienced user; can become complex if many menu options	Most general purpose systems
Form fill-in	Simple data entry; easy to learn; checkable	Takes up much screen space; causes problems where user options do not match the form fields	Ordering
Command language	Powerful and flexible	Hard to learn; poor error management	Operating systems, command and control systems
Natural language	Accessible to casual user; easily extended	Requires typing; NL understanding systems may be unreliable	Information retrieval and Q/A systems

Ian Sommerville: Software Engineering, 7th Ed., 2004 24

Implementer vs. User

User vs. Implementer

- **Thesis:** Engineer who implements something is seldom the right person to judge the goodness of the user interface
  - Knows how should work
  - Has a mental model of inner workings
  - Motivate to reduce implementation complexity
- Contrast user
  - Doesn't know how works – shouldn't have to?
  - Benefit from reduced use complexity
    - Reduced cognitive load

Foolproof quote

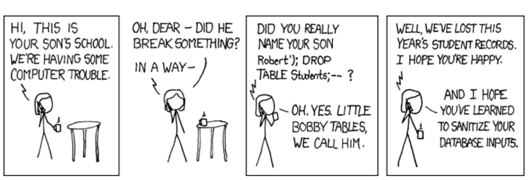
- **You cannot make something foolproof, because fools are so ingenious!**
  - George Cox

Issue

- **Hard to put aside what you know and see how it will look to an uninitiated user**
- **How could anyone not know?**
  - Naming a variable "foo-bar" might be interpreted as subtraction
  - "NC" means not connected
    - (user named their next state variables NA NB NC ND)
  - When program crashes, it leaves a lock file around that needs to be cleaned up…
- **Why would anyone**
  - Put a ' in a name?

Why would anyone

- **https://xkcd.com/327/**



Approach

What Can We Do With Principles?

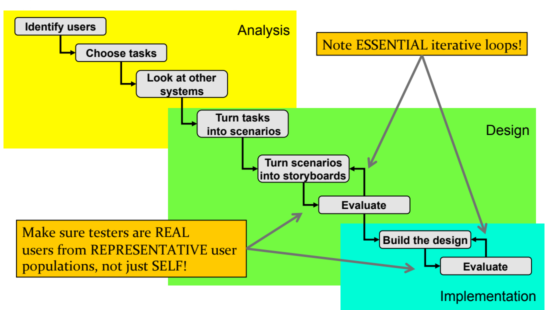
Principles are generally:

- **Descriptive, comparative and analytical (i.e., how alternatives compare; test and refine paradigm)**
- **Not constructive (i.e., do not define process of developing user interface design)**
  - No automated (good) interface design tools exist (e.g., that could have predicted the iPod user interface design)

Principles Must Be Considered in the Context of User Population

- **Principles define an optimization problem where the (target) user population is not uniform in skill, cognitive ability, needs, experience, learning style, or motivation.**
- **http://dilbert.com/strip/2008-12-10**

Task-Centered Design



http://www.comp.rgu.ac.uk/staff/sw/materials/cm2006/Interface%20design.L4.ppt

User Analysis

- **If you don't understand what the users want to do with a system, you have no realistic prospect of designing an effective interface.**
- **User analyses have to be described in terms that users and other designers can understand.**
- **Scenarios where you describe typical episodes of use, are one way of describing these analyses.**

Ian Sommerville: Software Engineering, 7th Ed., 2004

Analysis Techniques

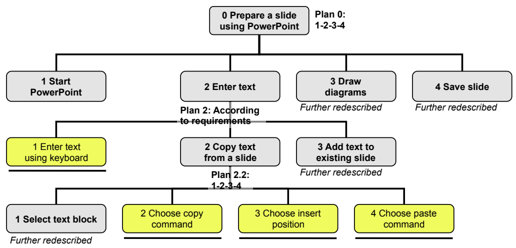
- **Task analysis**
  - Models the steps involved in completing a task.
- **Interviewing and questionnaires**
  - Asks the users about the work they do.
- **Ethnography**
  - Observes the user at work.

Ian Sommerville: Software Engineering, 7th Ed., 2004

Task Analysis

- **Need to determine what tasks must be accomplished.**
- **Understand user's intentions: What user *wants* (or needs) to do, not necessarily *how* to do it.**
- **Task analysis breaks problem into individual task components.**
- **Top-down task analysis.**
- **Tasks still need to be wrapped in a UI.**

Example Task Analysis: Using PowerPoint



http://www.comp.rgu.ac.uk/staff/sw/materials/cm2006/Interface%20design.L4.ppt

Interviewing

- Design semi-structured interviews based on open-ended questions.
- Users can then provide information that they think is essential; not just information that you have thought of collecting.
- Group interviews or focus groups allow users to discuss with each other what they do.

Ethnography

- Involves an external observer watching users at work and questioning them in an unscripted way about their work.
- Valuable because many user tasks are intuitive and users find these very difficult to describe and explain.
- Also helps understand the role of social and organizational influences on work.

Ethnographic Records

Air traffic control involves a number of control 'suites' where the suites controlling adjacent sectors of airspace are physically located next to each other. Flights in a sector are represented by paper strips that are fitted into wooden racks in an order that reflects their position in the sector. If there are not enough slots in the rack (i.e. when the airspace is very busy), controllers spread the strips out on the desk in front of the rack.

When we were observing controllers, we noticed that controllers regularly glanced at the strip racks in the adjacent sector. We pointed this out to them and asked them why they did this. They replied that, if the adjacent controller has strips on their desk, then this meant that they would have a lot of flights entering their sector. They therefore tried to increase the speed of aircraft in the sector to 'clear space' for the incoming aircraft.

Insights from Ethnography

- Controllers had to see all flights in a sector. Therefore, scrolling displays where flights disappeared off the top or bottom of the display should be avoided.
- The interface had to have some way of telling controllers how many flights were in adjacent sectors so that they could plan their workload.

Prototyping

User Interface Prototyping

- **The aim of prototyping is to allow users to gain direct experience with the interface.**
- **Without such direct experience,**
  - it is impossible to judge the usability of an interface.
- **Prototyping may be a two-stage process:**
  - Early in the process, paper prototypes may be used;
  - The design is then refined and increasingly sophisticated automated prototypes are then developed.

Paper Prototyping

- **Work through scenarios using sketches of the interface.**
- **Use a storyboard to present a series of interactions with the system.**
- **Paper prototyping is an effective way of getting user reactions to a design proposal.**

Prototyping Techniques

- **Script-driven prototyping**
  - Develop a set of scripts and screens using a UI design tool. When the user interacts with these, the screen changes to the next display.
- **Use PowerPoint as a substitute for an editable script.**
- **Visual programming**
  - Use a language designed for rapid development such as Visual Basic.
- **Internet-based prototyping**
  - Use a web browser and associated scripts.

User Interface Evaluation

- **Some evaluation of a user interface design should be carried out to assess its suitability.**
- **Full scale evaluation is very expensive and impractical for most systems.**
- **Ideally, an interface should be evaluated against a usability specification. However, it is rare for such specifications to be produced.**
- **Can evaluate against any of the "design principles" lists you wish to use.**

Sample Usability Attributes

Attribute	Description
Learnability	How long does it take a new user to become productive with the system?
Speed of Operation (use)	How well does the system response match the user's work practice and task requirements?
Robustness	How tolerant is the system of user error?
Recoverability	How good is the system at recovering from user errors?
Adaptability	How closely is the system tied to a single model of work?

Technology Change

Preclass 3

- **How many instructions should we be willing to execute to save a second of human time?**
  - Cost of second of human time?
    - Assume $300K/yr., 250 days/yr, 8 hours/day
  - Energy cost of instruction?
    - 300pJ/instruction, $0.12/KW-hr
  - Number of instructions cost same as human-second?
- **Important effects this ignores?**

49

Impact

- **Can afford to spend computation to bridge between natural user view (interaction) and underlying implementation view**
- **Energy/op has reduced over time**
  - Increasing this ratio
- Can afford to spend **more** computation now than in past

50

Rise of voice control

- **Siri**
- **Ok Google**
- **Alexa**
- **Voice Remote**
- **Locally recognize "wake words"**
  - Ship off to server farm for bulk speech recognition

51

Preclass 4

- **How GPS data ease data lookup for bus stop, schedule?**
- **Compared to what must do without GPS data?**

52

Context Awareness

- **Sense context**
  - Can reduce information need to explicitly gather from user
  - Prioritize/reorder data presented
    - Know more about likely common case
- **Other context examples?**

53

Sensors

- **Open up new input modes and interaction possibilities**

54

Natural(?) Input

- **Audio processing**
- **Vision, Radar**
- **Motion (e.g. fitbit, iWatch)**
- **Biometrics**
- **Coupled with signal processing, cheap computation**
- **Opportunity to take input from natural interactions**

Big Ideas

- **User Interface essential**
  - And worth designing carefully and deliberately
- **Implementer seldom a good judge of interface goodness**
  - Knows too much about how should work
  - Conflict of goals
- **View should match user goals, not internal design**
  - Spend computing cycles to bridge
  - Make simple, safe, intuitive

Lab Due

- **Note: Lab due Today (by midnight)**
  - Last day of classes (not have due during reading period)
  - ***Final office hours now to 8pm***

Final

- **Review: Monday 6-8pm in Towne 337**
- **Final: Wednesday 5/2 6pm in Moore 216**
  - Same Rules as midterm
    - Calculators allowed (work that out in advance)
    - Closed book, notes
  - 15% of grade
  - Comprehensive

Final Topics

Pre Midterm
- Data representation in bits
- Sounds waves
- Sampling
- Quantization
- Nyquist
- Lossy/lossless compression
- Common case
- Frequency domain
- Psychoacoustics
- Perceptual coding

Post midterm
- Combinational Logic
- Finite-State Machines
- Stored-Program Processors
- Processing Requirements
- Process Virtualization
- Persistent Storage
- File Systems
- Intellectual Property
- Networking
- User Interface

Reading

- *The Design of Everyday Things*, Donald Norman -- a classic book on design for usability (broader than just hardware and software)
- *The Inmates are Running the Asylum*, Alan Cooper -- a manifesto calling out computer/software industry for poor design
- *Set Phasers on Stun: And Other True Tales of Design, Technology, and Human Error*, Steven M. Casey -- a series of anecdotes (case-studies) on how bad design and interfaces can go wrong, perhaps even killing people.

Learn More @ Penn

- **Courses**
  - ESE543 – Human Factors Engineering

Topic	CIS	CMPE	EE	SSE
Analog Circuits		**ESE215**	**ESE215**	
Compress	**CIS121**	**CIS121**		
Nyquist, Fourier			**ESE224**, ESE325	**ESE224**, ESE325
Optimization	**CIS320**	(many)		**ESE204**
Digital Logic	**CIS240**	**CIS240, ESE370,** ESE532		
Processor	**CIS371**	**CIS371**		
OS	**CIS380**	**CIS380**		
File System	**CIS380, CIS121**	**CIS380, CIS121**		
IP		EAS545	ESE545	ESE545
Networking		**ESE407** or **CIS553**	ESE407	ESE407
Embedded		**ESE350 CIS441**	ESE350	ESE350
UI				ESE543

End